A Hybrid Intrusion Detection System against Botnet Attack in IoT using Light Weight Signature and Ensemble Learning Technique

Erukala Suresh Babu
National Institute of Technology Warangal
Mekala Srinivasa Rao
Lakireddy Balireddy College of Engineering Department of Computer Science and Engineering
Rambabu Pemula
Raghu Engineering College
Soumya Ranjan Nayak (nayak.soumya17@gmail.com)
Amity University https://orcid.org/0000-0002-4155-884X
Achyut Shankar
Amity University

Research Article

Keywords: Intrusion Detection, Ensemble Learning, Anomaly-based, Signature-Based, Botnet Attack

Posted Date: June 23rd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-905197/v1

Abstract

Internet of Things (IoT) plays a substantial role in the digital era of the information and intelligent Age. The use of interactive internet apps has opened up opportunities for increased threats to cyber security. Recently, botnets threats in IoT had become the most common cyber security threats. These threats provide malicious services and carry out phishing links on the internet. Consequently, an efficient intrusion detection system (IDS) is needed to detect these botnet attacks and unknown attacks with a low false-positive rate. Existing IDS methods detect these new attacks but require a high-precision detector. Most of the existing IDS uses either single machine learning or multiple classifiers that fail to detect unknown attacks and produce a high false-positive rate. This paper proposes a hybrid-based IDS that solves and detects unknown and novel attacks with low false-positive rates, better accuracy levels, and detection rates. This proposed work is deployed using two IDS methods in a two-staged manner. First, we modelled a Signature-Based Detector against DDOS attack for providing a better detection rate, early detection of known and low false-positive rates. Next, we modelled an Anomaly-Based Detector against DDOS attacks to achieve low false alarm rates, improved accuracy levels, and detected botnet and unknown attacks using the Machine Learning-based ensemble technique. Finally, we evaluated the performance using the confusion matrix on the classified data. We assessed the classifier performance based on detection rate, precision, accuracy, AUC score, and false-positive rates. The proposed hybrid technique provides a lower false-positive rate and better detection rate than the proposed model's classification technique.

1. Introduction

In today's digital world, everyone is using the internet to access data. People are using the internet not only for accessing data but also for their professional, social, and financial needs. There are millions of devices across the globe which is connected to the internet. Most businesses, institutions, banks, and research facilities for most daily activities depend on internet connection. This growing dependence on the internet has opened up avenues for increased cybersecurity threats and attacks. Unauthorized transfers out of the banking industry, loss of protected data, intensive compromise of operational systems, alteration of medical diagnostic results, Extortion-hard-to-please payments to avoid operational issues are some of the seriously increasing impacts. The use of interactive internet-based applications has opened up opportunities for increased threats by taking advantage of the vulnerabilities present in the network. Some internet threats include Botnet attacks, spamming, phishing, Distributed Denial-of-Service (DDoS attacks) [12], stealing data, etc. The malicious users use various techniques to exploit the device's vulnerabilities and compromise device security. Hence, there has to be a reasonable level of security for an organization's resources to protect them from internet attacks. Organizations all over the world employ firewalls for protecting their network from the outside network. However, when protecting a personal network from the outside web, a firewall cannot be fully efficient in providing security. The work of the firewall is to allow only permitted packets (the packets from trusted sources) to pass into the network. Yet, firewalls can, up to some extent, filter the incoming packets based on security policy but are not perfect in stopping modern internet attacks such as Botnet attacks. Therefore, an Intrusion Detection System (IDS) [4, 17] is used to provide security to the network. This IDS monitors the packets flowing in the network and analyses them to find attacks from the internet trying to compromise the system security. It also tries to find internal attacks trying to misuse system resources. This proposed work is deployed using two IDS methods in a two-staged manner. First, we modelled a Signature-Based Detector against DDOS attack for providing a better detection rate, early detection of known and low false-positive rates. Next, we modelled an Anomaly-Based Detector against DDOS attacks to achieve low false alarm rates, improved accuracy levels, and detected botnet and unknown attacks using the Machine Learning-based ensemble technique. Finally, we evaluated the performance using the confusion matrix on the classified data. We assessed the classifier performance based on detection rate, precision, accuracy, AUC score, and false-positive rates. The proposed hybrid technique provides a lower false-positive rate and better detection rate than the proposed model's classification technique.

Overview of Botnet Attack: Today, most smart devices use the internet to provide better service and help people make better decisions. However, the intruder is trying to steal the information, compromise the system privacy, damage the systems and exploit the system resources. One such type of attack is a Botnet Attack. A Botnets special case of DDoS attack [14] in which many devices are compromised and used to prevent the user of a single device from getting services, destroy data, send spam malware, and allow the devices and their link to be accessed by the attacker. A botnet is a group of robot software bots that is stored on an internet-connected device. A bot is a Bot Master-controlled infected host (a person or a group of people that control bots remotely). The bot-master will use Command and Control (C&C) software to manage these botnets. The botmaster takes advantage of the vulnerabilities on a compromised device that allows malicious content to be installed on devices without owners' knowledge. The target host installs the binary bot and approaches the Internet Relay Chat (IRC) server address by resolving the Domain Name Service. Then the victim enters the botmaster's IRC server to receive the commands. These botnets are different from any other malware because they follow the approach of Command-and-Control. When infected hosts receive commands, attack traffic overwhelms the victim's device, as shown in figure-1.

Recently, botnets threats in IoT had become the most common cybersecurity threats. These threats provide malicious services and carry out phishing links on the internet. Consequently, an efficient intrusion detection system (IDS) [8, 13, 15, 16] is needed to detect these botnet attacks and unknown attacks with a low false-positive rate. Existing IDS methods detect these new attacks but require a high-precision detector. Most of the existing IDS uses either single machine learning or multiple classifiers that fail to detect unknown attacks and produce a high false-positive rate.

Our Contribution: The proposed hybrid-based intrusion detection system solves the following above issues by deploying the two IDS techniques in a phased manner to identify the botnet and known attacks.

1. To model a Signature-Based Detector against DDOS attack for providing better high detection rate, early detection of known and low false-positive rate.
2. To model an Anomaly-Based Detector against DDOS attacks to achieve low false alarm rates and better accuracy and detect unknown attacks using Machine Learning based ensemble technique.

Table-1: Summary of Various Existing Methods on Intrusion Detection System

References	IDS deployment	Detection Methodology	Data Set	Implementation	Advantages	Disadvantages
Chien-Hau Hung et al	Distributed	Anomaly-based	Zeus, Waledac, and Virut	Yes	Detecting potentially infected bots by using machine learning, Feasible, expansible	A single weak classifier is used
Xuan Dau Hoang et al	-	Anomaly-based	Extracted and labeled domain name datasets	Yes	Effectiveness of the botnet detection model based on machine learning techniques	Attacks against DNS protocol are detected, Accuracy is just over 85%
Navjot Kaur et al	Centralized	Signature-based	Signature database	Yes	False alarm reductions Real-world application	Unable to identify the novel botnets. We need to update the knowledge base with new signatures
Faizal M et al	-	Anomaly-based	The dataset from the testbed environment	Yes	Compared various ML algorithms for better classification	Attacks against the only HTTP No attribute selection & reduction
P. Ioulianou et al	Distributed & Centralized	Signature-based	Cooja simulator	No	External threats & Internal compromised devices	Cannot detect unknown attacks Detects only flooding attack in RPL protocol
David Zhao et al	Online	Anomaly-based	Zeus botnet C&C and sample traffic from openpacket.org website.	Yes	Detecting bot activity in both the command and control and attack phases Novelty Detection	A single weak classifier is used Difficult to realize a full implementation of such a system on a large internal network

2. Related Work

This section presents the literature survey, which provides the basis for the proposed work. In [1] Xuan Dau Hoang et al. suggested using Domain Name Service (DNS) query data to detect the botnet based on machine learning techniques such as KNN, decision trees, random forest, and Naive Bayes. The model contains two stages, one training, and the other

detection. Domain names are collected from the data and pre-processed to extract the relevant attributes used to train the classifiers. In the detection phase, these trained classifiers are used to recognize the legitimate or botnet domains. The data sources are from virustotal.com and Alexa. The dataset is divided into four subsets, out of which three are used for training and one for testing. And different classification metrics, such as false rate, accuracy, etc., are used for evaluating the chosen learning techniques. The results showed that the random forest algorithm provides better performance when compared to values predicted by the remaining classifiers.

In [3] David Zhao et al. presented a novel method for detecting botnet activity focused on traffic behaviour analysis using machine learning to identify network traffic behaviours. Methods of traffic behaviour analysis don't rely on the packet's payload. The network traffic flow was examined for collecting the characteristics in smaller time intervals. For the classification purpose, attributes such as source and destination address (flow-based features), avg packet length, packets in the given time window, etc. The method consists of two phases. They had selected the REP tree machine learning algorithm. The classifier, the combination of known normal and malicious attribute vectors, is given as the input for training. The data sources are from the Honey project and Ericsson Lab. The results predicted by the model have a detection rate above 90% and a less false rate.

In [5] Navjot Kaur et al. analysed the network traffic of a particular organization for detecting peer-to-peer bots present in the network. They have used bot hunter that is developed on Snort, an open-source software. Internet traffic is passed on to the detection engine, which uses available signature rules for identifying malicious traffic and produces alerts and stores in the log files, which helps in future updates. Different botnets such as Phatbot, Napster, Ares, and Bit Torrent are detected.

In [7] P. Ioulianou et al. designed a framework for detecting the attacks which focused on IoT devices. They have developed both centralized and distributed Intrusion detector modules using a signature or misuse-based approach. They have created the testbed environment by introducing a Denial of Service (DoS) scenario of attacks on IoT devices using the Cooja simulator. The mitigated attacks are sinkhole, routing attacks using RPL protocol, and selective forwarding. The modules are incorporated with routers and detectors. For a group of sensors, one router through which all traffic is passed and many detectors are used. Routers run a detection module that compares the available signatures of IoT attacks. The firewall acts as an extra layer of defence by blocking suspicious IP addresses.

3. Proposed Work: Hybrid Based Intrusion Detection System

The section presents proposed work that solves the botnet attack in IoT using a hybrid-based intrusion detection system, which has better advantages-increased detection probability, a lower false alarm rate, and minimal detection delay over existing methods. First, we propose an approach to enhance detection accuracy using the AdaBoost ensemble learning detection technique [9,10] using various machine learning classifiers such as Support Vector Machine, Naive-Bayes, and Decision tree. This proposed ensemble technique provides a lower false-positive rate and better detection rate than the proposed model's classification technique. Next, we extended the proposed work with the misuse-based signature system that quickly identifies and detects the botnet intrusion early from the proposed anomaly-based systems. Finally, we evaluated the performance using the confusion matrix on the classified data. The performance of these classifiers is assessed based on false-positive rates, precision AUC scores, and accuracy levels.

3.1 Architecture of Hybrid Based Intrusion Detection System

Intrusion detection systems (IDS) are essential for defending the system in the context of rising security flaws. Traditionally, there are two types of IDS, namely- Anomaly-based detection and Signature-based detection. These IDS are essential for defending the system in the context of rising security flaws. In the Anomaly-based approach, normal data models are constructed depending on regular network traffic, and then the divergence from the standard model is

regarded as an attack or abnormal. This strategy has the key benefit of being able to track threats that compromise modern and unexpected vulnerabilities. However, these methods have the more remarkable ability of anomaly detectors to detect unknown or novel attacks. But it suffers from a significant defect that leads high false-positive rate. These defects occur due to the inadequacy of a training data collection covering all the legitimate aspects, and the other is that irregular activity is not always an intrusion indication. Another IDS is Signature-based detection that works by taking the stream of network packets or performing the audit trails based on signatures. These signatures are mainly used to detect the attacks by be identifying the behaviour of network traffic or analysing the audit trails. However, this signature-based detection cannot detect new or unknown attacks.

The proposed hybrid-based intrusion detection system combines signature-based and anomaly-based IDS advantages that can detect known attacks, botnet attacks, and unknown attacks. Combining these two approaches provides an effective IDS system that enhances the overall performance of botnet attack detection, low false alarm rates, high detection rate, and improved accuracy levels. As shown in figure-2, the proposed work is deployed using two IDS methods in a two-staged manner. In the first stage, misuse detection is employed using Snort IDS, a lightweight signature-based detector. This proposed Snort IDS maintains the database that contains the detection behaviour of known attacks and compares it with the network traffic of intrusion behaviour. If any abnormality is detected, the IDS system will generate alerts according to the event handling information present in the rules. So, the attacks are detected early without passing through further learning stages. In the second stage, we modelled an Anomaly-Based Detector against botnet attacks to achieve low false alarm rates, improve accuracy levels, and detect botnet and unknown attacks using the Machine Learning-based ensemble technique. This stage is mainly used to overcome the limitation of the first stage. After passing through signature-based detection, the remaining unknown network traffic is directed to the feature extraction stage to extract robust network features. The extracted non-redundant features are important and selected to discriminate abnormal behaviour from normal network activities. This process is achieved using machine learning classification such as Support Vector Machine, Naive-Bayes, and Decision Tree with the Adaboost Ensemble technique for increased accuracy in detecting malicious packets.

A. Signature-based Detector

The proposed misuse-based detection mechanism uses snort [7, 11], a popular lightweight signature-based IDS that can monitor the data flow in the network and analyze the network traffic. This IDS will generate the alerts, performs the protocol analysis, and finally detects different types of attack. In the IoT network, packets flow from the sensor node to the application passing through various network technology, internet technology, and service discovery processes. Snort is a network packet sniffer that inspects the content of the packets and identifying the behaviour of network traffic with known signatures using the rules, which is encapsulated within the signatures to detect the abnormal connections, record events, initiate action, and stores the related information in the database or log file. If the pattern matches and performance degrades, snort stops the packet processing, discards the packet, and stores its detail in the signature database. Finally, it compares those packets with the database of known attack signatures, and warnings will be generated with various attacks in the network, as shown in the figure-2. Snort IDS is consists of the following major components

- **Packet decoder**: The packet decoder collects packets from different network interfaces and then sent to the preprocessor or sent to the detection engine.
- **Preprocessors:** It modifies or arranges the packet before the detection engine to apply some operation on the packet if the packet is corrupted.
- **The Detection Engine:** Its work is to find out intrusion activity exits in a packet with the help of snort rules, and if found, then apply appropriate rules; otherwise, it drops the packet.

- **Logging and Alerting System:** Whatever detection engine finds in the packet, it might generate an alert or be used to log activity.
- **Output Modules:** Output modules or plug-ins save output generated by the logging and alerting system.

The proposed Snort IDS detects Volume Based DDoS Attacks that include the combination of HTTP, ICMP, UDP, and other spoofed-packet floods to target the victim for the resources. The attacker will randomly spoof the IP source, combine UDP packets with the port 80 for the destination, and send ICMP echo request packets that target the victim machine. The proposed Snort captures the UDP packets, ICMP packets, and HTTP packets recorded in the database and alerts the recorded different detected intrusions to the user.

B. Anomaly-based Detector

In the second stage, we modelled an Anomaly-Based Detector against botnet attacks to achieve low false alarm rates, improve accuracy levels, and detect botnet and unknown attacks using the Machine Learning-based ensemble technique. This stage is mainly used to overcome the limitation of the first stage. After passing through signature-based detection, the remaining unknown network traffic is directed to the feature extraction stage to extract robust network features. The extracted non-redundant features are important and selected to discriminate abnormal behaviour from normal network activities. This process is achieved using machine learning classification such as Support Vector Machine, Naive-Bayes, and Decision Tree with the Adaboost Ensemble technique for increased accuracy in detecting malicious packets.
The above figure-2 depicts the architecture of Anomaly-based detection that contains the Network Traffic, Feature Generation, Feature Selection, Classification Models and Alert/Warnings

- **Network Traffic:** In the behaviour-based identification approach, network traffic is monitored to identify any suspicious activity. For building the network-based detection system, the network traffic is taken from data source UNSW-NB15 [5]. Unlike other data sets, UNSW-NB15 [5] does not contain redundant records that affect detection biases and has a hybrid of the real modern normal and the contemporary synthesized attack activities of the network traffic. It includes statistical features such as flow-based, packet-based, content-based, time-based, which are used as input data for identifying the malware attacks.
- **Feature Generation:** Features are obtained from the network packets passing through layers of a standard Internet model. These attributes include a variety of packet-based features and flow-based features. The packet-based features examine the payload besides the headers of the packets.
- **Feature Selection:** It is the process of selecting the attributes that can make the predicted outcome more accurate. The selection of features plays a crucial part in the detection system for choosing relevant features and eliminating unwanted redundant records. These decrease the training time of the learning model and affect the general performance of any intrusion detection system. The selection of features aims at reducing the cost of computation involved, removing redundant information, improving accuracy, and helping to analyse the network data. The Correlation Coefficient measure was applied to look at the relation between the attributes and variables of the data and collect the most vital features.
- **Classification Models:** This model is used to classify the legitimate and anomalous feature vectors from the dataset. These datasets contain flow-based, packet-based content of network traffic. There is a need to perform the numerical statistical characteristics such as mean (for every 100 packets) and size of the packet, protocol information with the direction identifiers. The following classification techniques have been used to classify network records with moderate variations between their regular and malicious observations.

I. Support Vector Machine: The Support Vector Machine or SVM is one of the Supervised Learning Algorithms used for problems with classification and regression. This algorithm aims to build a decision boundary that could segregate n-dimensional space into groups. So in the future, we can conveniently put the new data set into the appropriate class. This

boundary to the decision is called a hyperplane. The main advantage of this approach is that (1) it is efficient in spaces with significant dimensions. (2) It is still accurate in instances if the number of samples present in the dataset is less than the number of dimensions of the sample. (3) It is effective in memory as it selects a set of training samples known as support vectors in the decision-making function.

II. Naive-Bayes: This classifier that works on Bayes probability theorem (conditional probabilities) and assumption of attribute independence in a given data, i.e., altering the value of an attribute, will not change the value of the other features. The above technique fits well for large data sets and is, therefore, better adapted for real-time predictions. The main advantage of this approach is that (1) Implementation is easy and quick. (2) It is highly flexible and needs less data. (3) It makes predictions using probabilities. (4) It can handle both discrete and continuous data. (5) It can efficiently deal with missing values. (6) Easy to update as new data arrives.

III. Decision Tree: The decision tree structure is similar to a flowchart or rooted directed tree. The leaves are called nodes, and branches are called edges. The nodes with no outgoing edges are called terminal nodes that contain class labels. All other nodes are called internal nodes or child nodes constructed by a series of if…. then… rules that classify the given input data. We selected the C4. 5 technique from the available decision tree algorithms to construct quite predictive models, including continuous data and incomplete data (missing attribute values). It handles both classification and regression problems. This technique uses Information gain measurement for splitting. An n dimensional relational data containing attribute values and respective class labels are required for training the model. The main advantage of this approach is that (1) it allocates a particular value to each decision, problem, and outcome(s). (2) It decreases complexity and confusion and also improves clarity. (3) It considers any possible outcome of a decision into account and consequently tracks each node to the conclusion. This technique is simpler and more efficient because there are no complicated equations or data formats needed.

We analysed and compared the performance and accuracy of the individual classifiers. But the performance of each classifier is showing poor results. An Ensemble Machine Learning is proposed to achieve better performance, which combines all the classifier techniques as shown in figure-3.

IV. Ensemble Machine Learning Approach: This approach is mainly used to achieve high accuracy using Adaptive Boosting (Adaboost) Classifier. This classifier is used to improve classifiers accuracy, which is an iterative ensemble method as shown in figure-4. This classifier combines Support Vector Machine, Naive Bayes, and decision trees make them strong classifiers, improves their accuracy, creates high precision models, and will be less affected by the overfitting problem. The primary objective of this method is to allocate weight in training set for each instance. Initially, all weights are considered to be equal. Still, the weights are raised for all instances predicted wrongly in each iteration such that in the next epoch these instances are given a high likelihood of classification. In contrast, the weights of correctly classified instances are reduced. The iterations are repeated until a good classifier with a low error rate is reached or until we exceed the defined maximum number of estimators. Each iteration reduces training error and tries to ensure a good fit for the data given.

As shown in figure-3 ***Voting Classifier*** is used to get the final prediction from the above three strong classifiers. Soft voting is applied to the outcomes of the classifiers. Soft voting is achieved by averaging the probability distributions estimated by the individual techniques to the best result. We predict the class labels based on the calculated classifier probabilities and the assigned weight to the classifier in soft voting.

V. Alert/Warnings: When malicious instances are classified, alerts are generated, and the false positive rate is found for evaluating the performance.

4. Results And Performance Evaluation

This section presents the results and performance evaluation of the proposed system. We evaluated and compared the performance of the experimental results of the proposed hybrid IDS that combines the Signature-based and Anomaly-based detectors using various parameters such as Accuracy, Precision, Detection Rate, False Positive Rate (FPR), F1_score, etc. To evaluate the performance of the experimental results of our proposed framework, we utilize the *UNSW-NB15 [5] dataset* benchmark information that was made by the IXIA Perfect Storm device in the Cyber Range Lab of the Australian Center for Cyber Security (ACCS). This data source has a hybrid network traffic's actual modern normal activities and contemporary synthesized attacks. The labeling of normal vectors is given class as Normal, and for malicious vectors, the class is labeled as an attack. The tool tcpdump was also used to acquire all the raw network packet data, and features were created with the tools Argus, Bro-IDS, and twelve algorithms to generate 43 features with the class label. The nominal data type of class label of each record of the data is assigned with numerical values such as for normal instances as 0 and attack as 1. The data were separated into two sets: a training set and a test set in the 7:3 ratio, with approximately 175,217 training records and 82,456 test records. Each classifier was trained using a train set and validated using the test set. The underneath Table-2 gives normal and attack records in the dataset used for training the model.

Table-2: Proposition of normal and attack records used for training the proposed model.

Traffic Type	Training	Testing	Total
Normal	1,19,341	45,332	**1,64,673**
DDoS Attack	56,000	37,000	**93,000**
Total	**1,75,431**	**82,332**	**2,57,673**

In the *UNSW-NB15* dataset, the categorical features with nominal data type are the following attributes such as service à (https, FTP, DNS, ssh…), Protocols à (UDP, TCP, and ICMP), State à (TST, URN, RTA…..). These categorical fields are encoded into numerical data types, with each being of unique value such as TST=1, URN=2, RTA=3, and so on. But the *UNSW-NB15 dataset* is not ready to fit into the chosen classification models. It contains both quantitative and qualitative features that may have unwanted and redundant features, which cannot be used for the statistical techniques models. However, the proposed machine learning techniques use numerical statistics for classifying the given input data that gets affected by the qualitative attributes present in the data source. The quality of input data plays an essential role in obtaining a well-trained machine learning model. Hence, data needs to be visualized for its quality and needs to be pre-processed before training the proposed model; the data should be cleaned and prepared for fitting into the used classification models. The features in the dataset are visualized for correlation. If any two attributes are highly correlated to each other, they produce the same effect on the dependent variable. To reduce the unnecessary computation or any other costs, we can discard one of the two attributes. To perform, we have used matplotlib, pandas packages for plotting the data correlation. The features are ranked within the [-1, 1] interval as shown in the below figure-5. The figure-5 also shows most of the dataset features that have correlation values in the interval [-0.4, 0.4], which gives us that the attributes are moderately correlated or have very little correlation. Hence, these features can be used for training the machine learning classifiers.

4.1 Performance Metric

The proposed work uses the performance metrics to assess Signature-based and Anomaly-based detectors using machine learning techniques to measure the efficiency. The differing metrics used include precision, accuracy, recall or detection rate, false-positive rate, F1- score. The assessment of the proposed Intrusion Detection System (IDS) is assessed based on the following measurement

i. **Accuracy** is calculated as the proportion of the adequately classified samples to the total samples.
ii. **Precision** is determined as the fraction of true positive samples to predict positive samples. It is the assurance of detection of a DDOS attack.
iii. **Recall** is expressed as the fraction of true positive samples to total positive samples and referred to as Detection Rate (DR) or True Positive Rate (TPR).
iv. **False-Positive Rate (FPR)** is determined as the percentage of false-positive samples to the positive samples predicted.
v. **F1_Score (F1)** is described as the precision and the recall harmonic average.
vi. **Area Under the Curve (AUC):** The sum of the region under a Receiver Operating Characteristic (ROC) curve, a plot of the False-Positive Rate in the X-axis and the True-Positive Rate in the Y-axis, presenting the complete performance of a model.

The above performance measures are obtained from the confusion matrix based on the predicted class calculated versus the actual class (ground truth). The confusion matrix is the process of presenting the result of binary classification. There are four possible outcomes as follows based on the two-class nature of the prediction:

- True-Positive (TP): Number of Attacks/anomalies that are successfully detected as attacks.
- False-Positive (FP): Number of Normal records that are incorrectly classified as attacks.
- True-Negative (TN): Number of Normal records that are successfully identified as normal.
- False-Negative (FN): Number of Attacks/anomalies that are classified as normal.

In phase one, we evaluated the performance of the experimental results of the Signature-based using Snort IDS. We tested a total of 82,332 packets of UDP, ICMP and, HTTP Packets. These UDP, ICMP, and HTTP Packets are captured by snort based on the rules/signatures. These rules/signatures are written DDoS attacks to detect intrusions. All the alerts are generated from snort IDS that are logged into the output module. All these modules are tested, and results are achieved.

Table-3 Comparison of Performance Metrics using Snort IDS

Proposed Method	Measures (%)			
	Packets	ACC	DR	FPR
Snort IDS	25,000	98.67	88.56	0.7
	50,000	98.70	90.72	0.9
	82,332	98.17	89.43	1.3

The table-3 shows the significant in Detection Rate 88.56% for 25,000 packets, 90.72% for 50,000 and, 89.43% for 82,332 packets respectively. It can also observed from the table-2 that the proposed IDS provides low false-positive rate 0.7% for 25,000 packets, 0.9% for 50,000 and, 1.3% for 82,332 packets respectively and finally the proposed IDS achieves high accuracy 98.67% for 25,000 packets, 98.70% for 50,000 and, 98.17% for 82,332 packets respectively.

Table-4 Comparison of Performance Metrics using Anomaly-based detector

Machine Learning Algorithms	Measures (%)					
	ACC	PRE	DR	F1	FPR	AUC
DT	93.7	95.0	95.0	95.0	8.9	93.2
NB	76.6	79.1	86.0	82.4	37.9	84.1
SVM	74.1	73.5	65.3	76.2	13.5	78.0
DT+NB	94.2	94.7	95.7	95.2	9.3	98.0
NB+SVM	76.7	75.3	92.4	84.0	54.5	69.8
DT+SVM	94.1	94.7	95.9	95.2	9.4	97.0
DT+NB+SVM	94.3	94.8	96.3	95.5	9.0	98.0

The UNSW-NB15 dataset is divided into training and testing subsets to evaluate each classifier and Ensemble Method. Table-4 and figure-6, Figure-7, and figure-8 show the DT technique produces a 93.7% accuracy, 95.0% detection rate, and 8.9% FPR, the SVM technique achieves a 74.1% accuracy 65.3% detection rate, and 13.5% FPR. Lastly, the NB technique achieves an accuracy rate of 76.6 %, 86.0% DR, and 37.9% FPR. When DT and NB techniques are combined with an accuracy rate of 94.2%, the detection rate of 95.7% and 9.3% FPR are achieved. NB and SVM techniques combined achieved an accuracy of 76.7 %, 92.4% DR, and 54.5% FPR. Similarly, when DT and SVM techniques are combined with an accuracy rate of 94.1%, the detection rate is 95.9%, and 9.4% FPR is achieved. The Accuracy and Detection of the Ensemble method of the three techniques achieve 94.3 % and 96.3%, respectively, while the FPR produced is 9.0%, which outperforms the DT, NB, and SVM performance techniques. The figure-6, Figure-7, and figure-8also shows the the significant in Detection Rate 88.56% for 25,000 packets, 90.72% for 50,000 and, 89.43% for 82,332 packets respectively. It can also observed from the table-2 that the proposed IDS provides low false-positive rate 0.7% for 25,000 packets, 0.9% for 50,000 and, 1.3% for 82,332 packets respectively and finally the proposed IDS achieves high accuracy 98.67% for 25,000 packets, 98.70% for 50,000 and, 98.17% for 82,332 packets respectively.

The figure-9 shows the receiver operating characteristic (ROC) curves; it is observed that the AUC score of Naive-Bayes is 0.841, and the Decision Tree has a 0.934 AUC score. The AUC score of the Support vector machine is 0.798. The AUC score of the proposed ensemble approach is 0.977, which shows this result proves the effectiveness of our proposed detection mechanism.

5. Conclusions

When Internet use is rising rapidly, the probability of attack in that ratio is also increasing. Botnets are one of the most significant threats to cybersecurity that companies face today. In this work, this proposed work is deployed using two IDS methods in a two-staged manner. This proposed work is deployed using two IDS methods in a two-staged manner. First, we modeled a Signature-Based Detector against DDOS attack for providing a better detection rate, early detection of known and low false-positive rates. Next, we modeled an Anomaly-Based Detector against DDOS attacks to achieve low false alarm rates, improved accuracy levels, and detected botnet and unknown attacks using the Machine Learning-based ensemble technique. We further evaluated the performance of each classifier. We combined these three classifiers in different combinations, and the Adaptive Boosting ensemble technique was applied. We tested the performance of the proposed framework using the UNSW-NB15 dataset by visualizing the relation between features in the data source by applying the Correlation Coefficient to remove redundant features. The proposed methodology is evaluated based on Detection rate, false-positive rate, accuracy, and AUC scores. Our results showed the proposed framework is better when compared to the results obtained from individual classifiers.

Declarations

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

This Research Received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Conflicts of interest/Competing interests

The author confirms that there is no conflict of interest to declare for this publication.

Availability of data and material

Data and material will be available based on the request

Code availability

Not Applicable

Authors' contributions

Erukala Suresh Babu: Formal analysis, Resources, Validation, Writing - original draft. **Mekala Srinivasa Rao:** Data curation, Investigation, Software, Writing - original draft. **Rambabu Pemula:** Data curation, Investigation, Software, Writing - original draft. **Soumya Ranjan Nayak:** Conceptualization, Methodology, Software, Writing - review & editing, Visualization, Supervision. **Achyut Shankar:** Investigation, Writing - review & editing, Supervision, and Validation.

CONFLICT OF INTEREST

The author confirms that there is no conflict of interest to declare for this publication.

References

1. Hoang, Xuan Dau, and Quynh Chi Nguyen. "Botnet detection based on machine learning techniques using DNS query data." Future Internet 10.5 (2018): 43.
2. Dollah, Rudy Fadhlee Mohd, et al. "Machine learning for HTTP botnet detection using classifier algorithms." Journal of Telecommunication, Electronic and Computer Engineering (JTEC) 10.1-7 (2018): 27-30.
3. Zhao, David, et al. "Botnet detection based on traffic behavior analysis and flow intervals." computers & security 39 (2013): 2-16.
4. Nour Moustafa and Jill Slay, "UNSW-NB15: A Comprehensive Data set for Network Intrusion Detection Systems (UNSW-NB15 network data set)," 2015 Military Communications and Information Systems Conference (MilCIS), pp. 1–6, 2015.
5. Kaur, Navjot, and Sunny Behal. "P2p-bds: Peer-2-peer botnet detection system." IOSR Journal of Computer Engineering 16.5 (2014): 28-33.
6. Ioulianou, Philokypros, et al. "A signature-based intrusion detection system for the Internet of Things." Information and Communication Technology Form (2018).
7. Dr.Prakash Sangwan and Vinod Kumar, "Signature-based Intrusion Detection System using Snort", International Journal of Computer Applications & Information Technology Vol. I, Issue III,(ISSN: 2278-7720).

8. M.Ali Aydin, A.Halim Zaim and K.Gokhan Ceylan, "A hybrid intrusion detection system design for computer network security" Computers and Electrical Engineering Volume 35, Issue 3.
9. Rokach, Lior. "Ensemble methods for classifiers." Data mining and knowledge discovery handbook. Springer, Boston, MA, 2005. 957-980.
10. Zhou, Zhi-Hua. "Ensemble learning." Encyclopedia of biometrics 1 (2009): 270-273.
11. Elshafie, Hussein M., Tarek M. Mahmoud, and Abdelmgeid A. Ali. "An Efficient Snort NIDSaaS based on Danger Theory and Machine Learning." Appl. Math 14.5 (2020): 891-900.
12. Merouane, Mehdi. "An approach for detecting and preventing DDoS attacks in campus." *Automatic Control and Computer Sciences* 51.1 (2017): 13-23.
13. Cepheli, Özge, Saliha Büyükçorak, and Güneş Karabulut Kurt. "Hybrid intrusion detection system for ddos attacks." *Journal of Electrical and Computer Engineering* 2016 (2016).
14. Buchanan, Bill, et al. "A methodology to evaluate rate-based intrusion prevention system against distributed denial-of-service (DDoS)." *Cyberforensics 2011* (2011).
15. Khamphakdee, Nattawat, Nunnapus Benjamas, and Saiyan Saiyod. "Improving Intrusion Detection System Based on Snort Rules for Network Probe Attacks Detection with Association Rules Technique of Data Mining." *Journal of ICT Research & Applications* 8.3 (2015).
16. Aickelin, Uwe, Jamie Twycross, and Thomas Hesketh-Roberts. "Rule generalisation in intrusion detection systems using SNORT." *International Journal of Electronic Security and Digital Forensics* 1.1 (2007): 101-116.
17. Bakhoum, Ezzat G. "Intrusion detection model based on selective packet sampling." *EURASIP Journal on Information Security* 2011.1 (2011): 1-12.

Figures

Figure 1

Botnet Attack Scenario

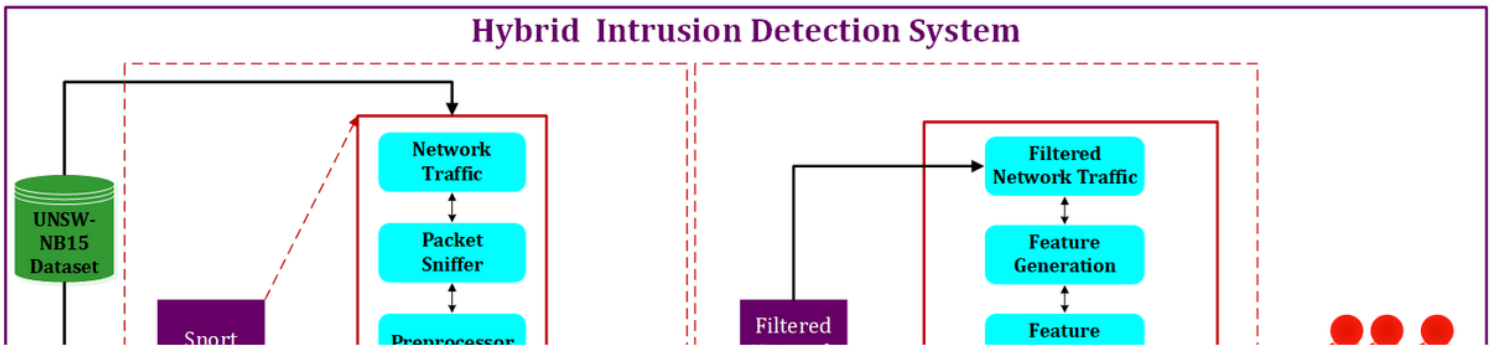

Figure 2

Proposed Architecture of Hybrid based Intrusion Detection System

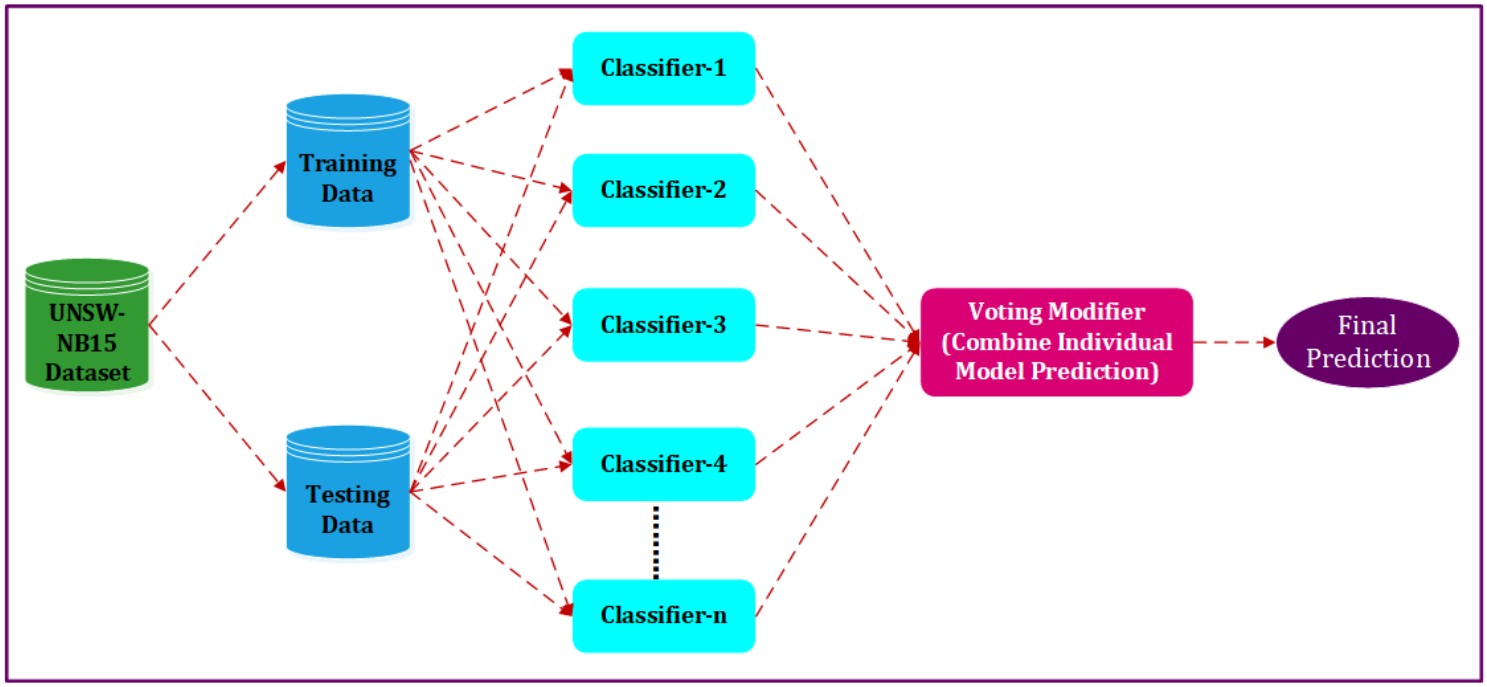

Figure 3

Ensemble Machine Learning Technique

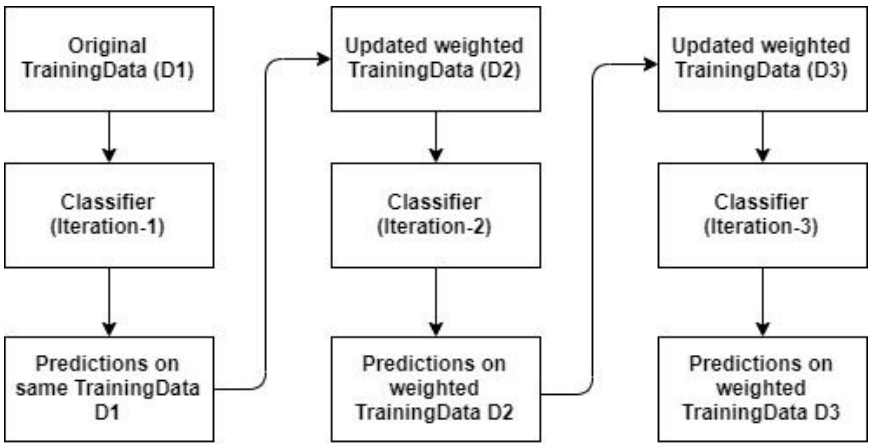

Figure 4

Iterations in Adaptive Boosting

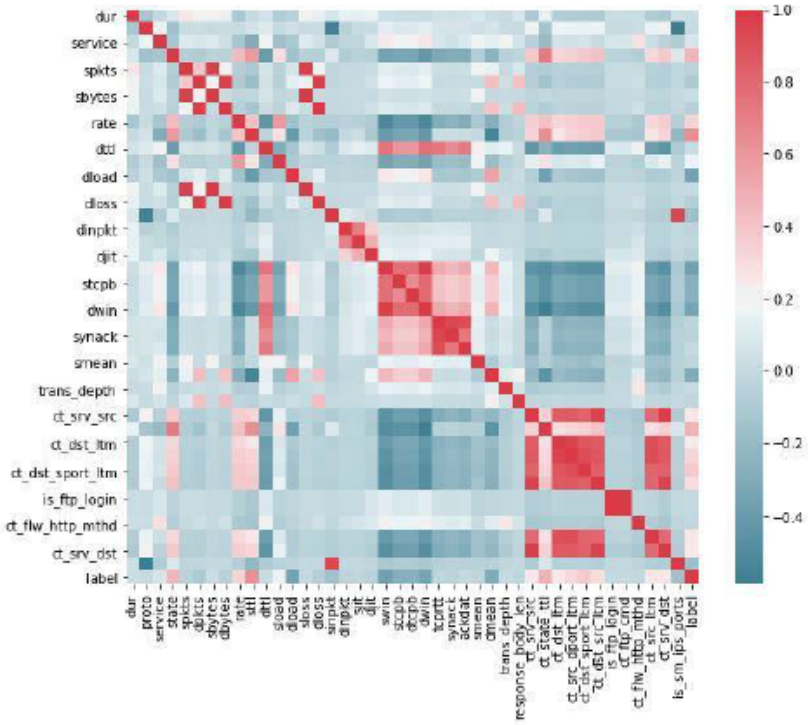

Figure 5

Correlation Coefficient of features

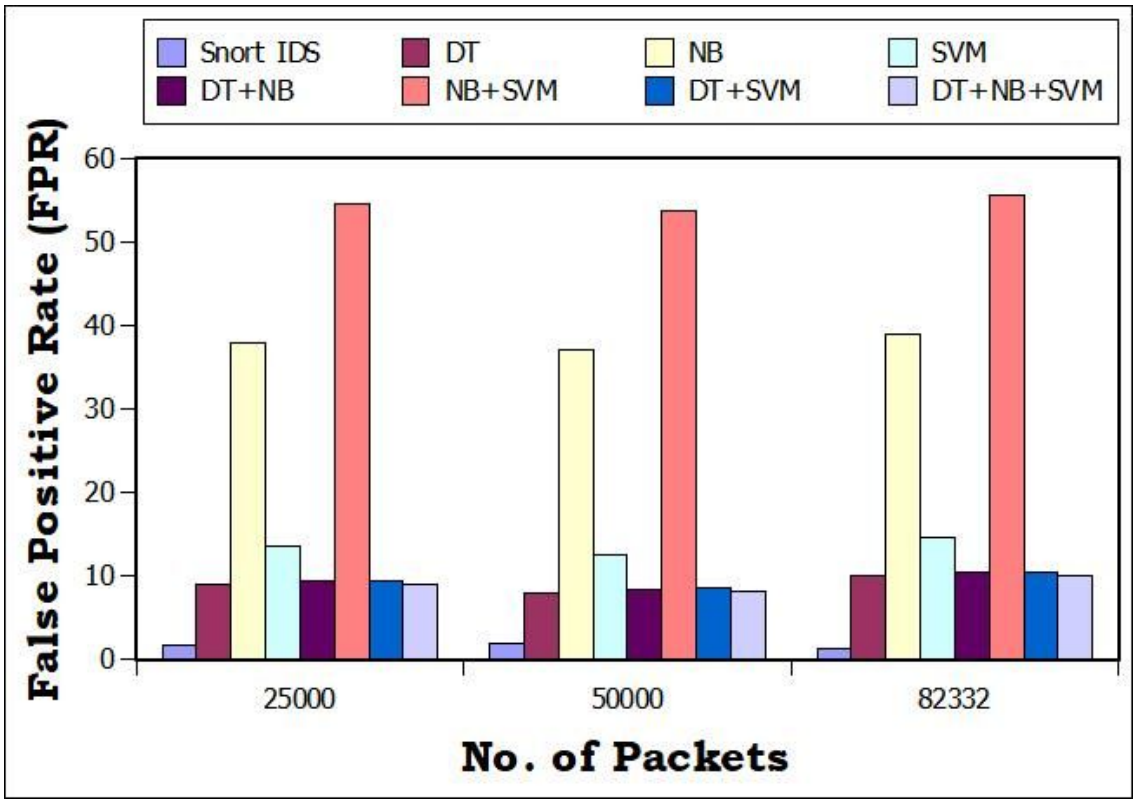

Figure 6

False-Positive Rate using Hybrid IDS

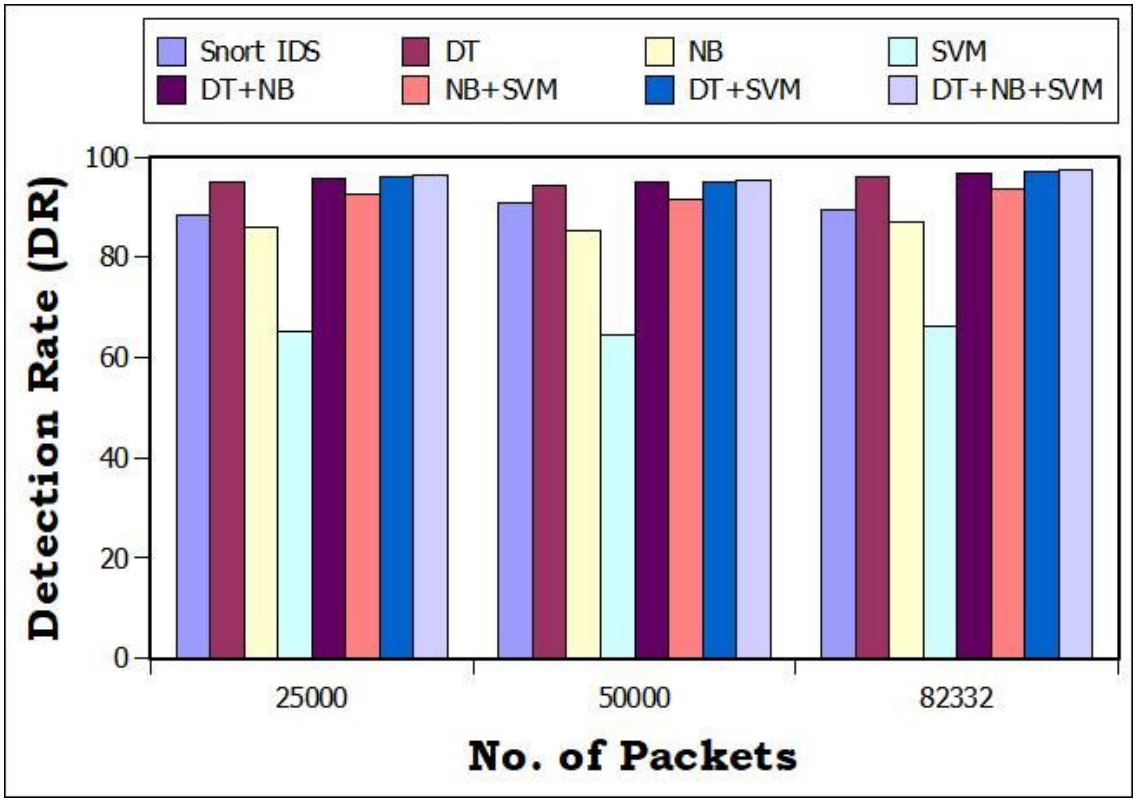

Figure 7

Detection Rate using Hybrid IDS

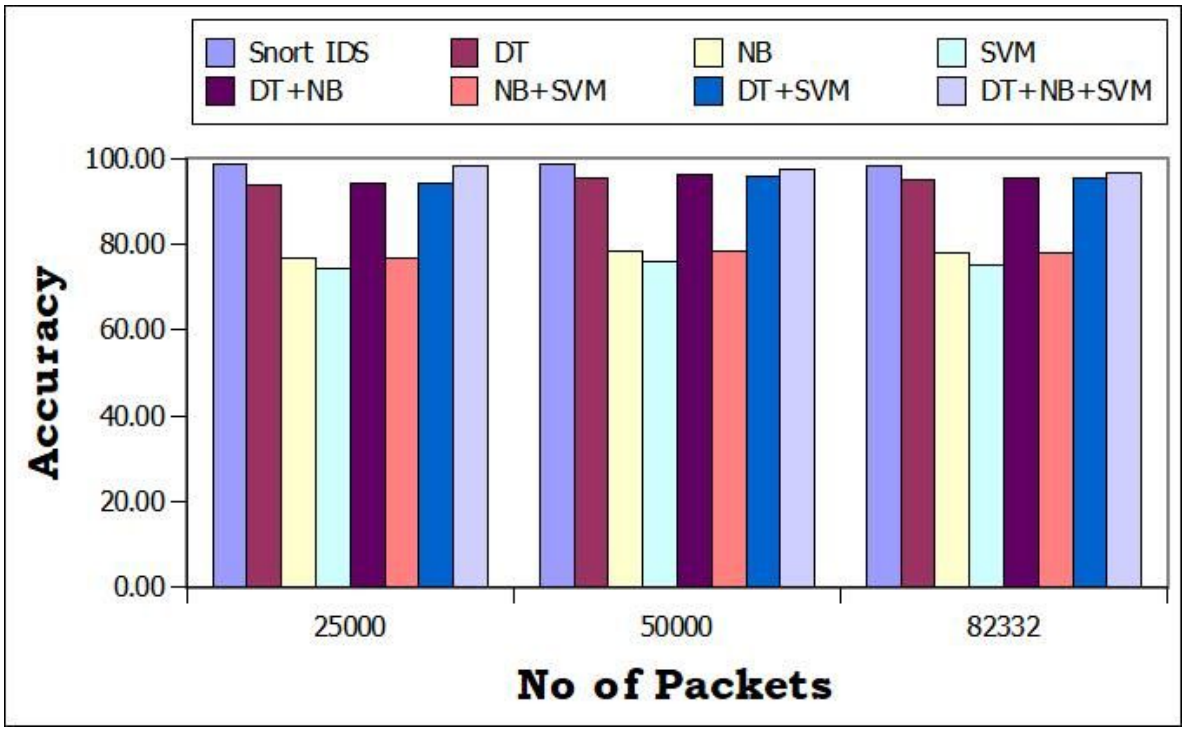

Figure 8

Accuracy using Hybrid IDS

Figure 9

ROC Curves and AUC scores of Proposed Anomaly-based detector