Novel Approach for IP-PBX Denial of Service Intrusion Detection Using Support Vector Machine Algorithm

Abdirisaq M. Jama[1], Othman O. Khalifa[2] and Nantha Kumar Subramaniam[3]

[1]Cluster of Applied Sciences, Open University Malaysia. Kuala lumpur, Malaysia
[2]Department of Electrical and Computer Engineering, International Islamic University Malaysia
[3]Cluster of Applied Sciences, Open University Malaysia. Kuala lumpur, Malaysia

***Abstract*:** Recent trends have revealed that SIP based IP-PBX DoS attacks contribute to most overall IP-PBX attacks which is resulting in loss of revenues and quality of service in telecommunication providers. IP-PBX face challenges in detecting and mitigating malicious traffic. In this research, Support Vector Machine (SVM) machine learning detection & prevention algorithm were developed to detect this type of attacks Two other techniques were benchmarked decision tree and Naïve Bayes. The training phase of the machine learning algorithm used proposed real-time training datasets benchmarked with two training datasets from CICIDS and NSL-KDD. Proposed real-time training dataset for SVM algorithm achieved highest detection rate of 99.13% while decision tree and Naïve Bayes has 93.28% & 86.41% of attack detection rate, respectively. For CICIDS dataset, SVM algorithm achieved highest detection rate of 76.47% while decision tree and Naïve Bayes has 63.71% & 41.58% of detection rate, respectively. Using NSL-KDD training dataset, SVM achieved 65.17%, while decision tree and Naïve Bayes has 51.96% & 38.26% of detection rate, respectively. The time taken by the algorithms to classify the attack is very important. SVM gives less time (2.9 minutes) for detecting attacks while decision tree and naïve Bayes gives 13.6 minutes 26.2 minutes, respectively. Proposed SVM algorithm achieved the lowest false negative value of (87 messages) while decision table and Naïve Bayes achieved false negative messages of 672 and 1359, respectively.

***Keywords*:** Voice over IP; Session Initiation Protocol; Attack; Security; Denial of Service, Support Vector Machine.

1. Introduction

We have seen an increase in malicious attacks on the internet over the past few years as the internet continues to grow and integrate more facets of our everyday life than ever. These attacks are mostly targeted towards communications, payments, and many other aspects [1]. Therefore, the importance for network security professionals to effectively identify these different types of attacks and to prevent them from using various network security techniques runs constant.

Voice over Internet Protocol (VoIP) is technology that uses connectivity over Internet Protocol (IP) networks to communicate with the system. In addition to traditional phone services including VOIP, it offers voice call flexibility and efficiency like that of the traditional Public Switched Telephone Network (PSTN).

If we compare VoIP to traditional telephony, it has emerged as a standard for voice communication using the Internet and it allows the integration of more communication options and at lower cost compared to traditional telephony.

A lot of interest has been devoted to strengthening the network of SIP without considering security of the protocol. SIP-based VOIP network can be prone to IP attacks. People should be aware of the different types of SIP attacks and countermeasures to overcome them. SIP based VoIP security issue has been met with a range of solutions and strategies. The relevance of a SIP-based VoIP communications security is well known among cyber security experts. There has been little research regarding DoS-based SIP security to this point in time.

This paper is based on an assessment approach that depends on the use of a Real Network topology. To the authors' knowledge, most of the publications reviewed do not use this type of deployment for evaluation. Testing DoS attacks on an operational network enables the most realistic testing environments.

The second section presents Intrusion Detection Systems (IDS). The third section explains Anomaly detection techniques using Machine learning classifiers. Section 4 addresses the proposed machine learning classifier to combat attacks in SIP based VoIP followed by its performance metrics. This section also illustrates related work to benchmark the performance of proposed algorithm and Section 5 concludes the paper providing some pointers to future research work.

2. Intrusion Detection System (IDS)

Denial of Service (DoS) attacks seek to make a server or system unavailable to its intended users [2].

An IDS is a system that monitors incoming and outgoing traffic to detect violations in the design. IDS could be a software or hardware system which detects malicious measures on computer systems to ensure system security.

Intrusion can be characterized as any malicious behavior causing information system harm. Any attacks that may pose a threat to the privacy, credibility or availability of information would therefore be considered an intrusion.

For instance, behavior that would prevent legitimate users from reacting to computer services is regarded as an intrusion.

The right attack detection phase should be in a good defense system before any reaction. Any system to detect attacks is intended to detect intrusions before significant harm can occur. Any unauthorized attempt to view, disprove, alter, or damage information to make a network unsatisfactory is also called Intrusion [3].

In a short period, a good system can detect attacks with a low proportion of false positives. Due to the rising number of intrusions, studies over the years working to introduce Intrusion Detection Systems (IDSs). The aim of the IDS is to identify different types of malicious network traffic and server usage that cannot be identified by a traditional firewall.

This is vital to ensuring a high level of protection against actions that compromise the availability, integrity, or confidentiality of server systems. The IDS systems can be broadly grouped into two types, The Signature-based Intrusion Detection System (SIDS) and the Anomaly-based Intrusion Detection System (AIDS) as explained in the next subsections.

2.1 Signature-based Intrusion Detection System (SIDS)

Signature intrusion detection systems (SIDS) are based on pattern matching techniques to find a known attack; these are also known as Knowledge-based Detection or Misuse Detection [4]. In SIDS, matching methods are used to find a previous intrusion. In other words, when an intrusion signature matches with the signature of a previous intrusion that already exists in the signature database, an alarm signal is triggered.

For SIDS, host logs are examined to classify sequences of previously defined commands or actions as malware. However, SIDS has trouble detect the zero-daily attack, since no matching signature remains in the database before the latest attack signature has been retrieved and stored. SIDS typically provides an excellent detection precision for documented intrusions.

2.2 Anomaly Intrusion Detection System (AIDS)

Signature The ability of anomaly intrusion detection systems to improve upon Signature based Intrusion Detection Systems (SIDS) has resulted in interest from many scholars. An intelligent computer system is created using machine learning, statistical-based or knowledge-based methods in AIDS. Any discrepancy between what the data shows and what the model predicts is an anomaly which requires explanation. Also, AIDS is an effective way of discovering internal fraudulent behaviors.

We assume that abnormal user behavior differs from how the typical user behaves. The behaviors of abnormal users which are dissimilar to standard behaviors are classified as intrusions.

AIDS research progresses in two phases: The training phase and the testing phase. In the training phase, the normal traffic profile is used to learn a model of normal behavior, and then the system's performance is measured on a new simulated data set. Due to classification into several categories based on the training method, AIDS can be classified into different categories as explained in the next section three [5].

3. Machine Learning

The Machine learning is a process for extracting knowledge from vast amounts of data. Machine learning models involve the application of a set of rules, methods, or complex “transfer functions” that can be applied to discover or identify similar trends [6].

In the past few years, machine learning methods have been used to analyze patterns in historical time-series data. One of the solutions to the problem of timely attack detection is to develop a classifier based on machine learning, that would ascertain if the incoming traffic has been under threat.

There are three well-known learning types in machine learning as follows [7].: -

• Supervised
• Unsupervised
• Reinforcement learning

Surveyed learning-based IDS strategies use the labeled training data to detect intrusions. This paper is based on Supervised learning.

The supervised approach to learning usually consists of two stages, namely training and testing. Relevant features and classes are defined during the training phase, and the algorithm will then learn from these data samples. Each record in supervised learning IDS is a pair containing a network or host data source and an associated output value, namely intrusion or regular output.

The supervised learning technique is then used to train the classifier using the training data for selected features to learn the inherent relationship between the input data and the labeled output value. During the test phase, the trained model is used to classify the unknown data into an intrusion or a regular class.

The resulting classifier will then become a model that predicts the class to which input data may belong, given the set of values of the feature. Figure 1 demonstrates a general method for applying the techniques for classification.

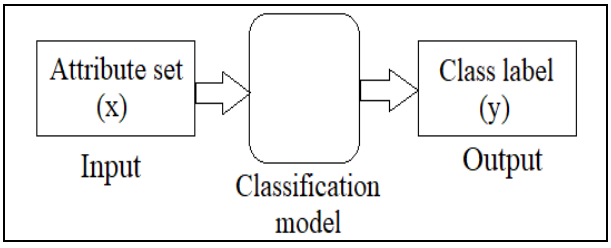

Figure 1. General classification technique

The performance of the classifier in its ability to predict the correct class is measured in terms of several metrics as discussed in Section 4.

There are several classification methods, such as decision trees, support vector machines and Naïve Bayes. Each technique uses a learning method to create a classification model.

However, the training data should not only be treated with an acceptable classification method but should also properly classify the class of records that it has never seen before. It is a key task of the learning algorithm to construct classification models with a reliable generalization capability.

3.1 Decision Tree (DT)

Decision tree consists of three basic components. The first component is a decision node that is used to identify the test attribute. The second branch is a branch, where each branch represents a possible decision based on the value of the test attribute. The third is a sheet comprising the class to which the instance belongs [8].

3.2 Naïve Bayes

This approach relies on the use of the Bayes principle among attributes with robust independence assumptions. Naïve Bayes answers questions like "how is there a probability, given the system activities observed, that a special type of attack will occur? "With the application of the formula for

conditional probability. Naïve Bayes reliance on the characteristics of attacks and normal behavior, which are different in probability. Naïve Bayes is among the most prevalent IDS classification model due to its simplicity of use and the efficiency of calculation both derived from its conditional autonomy [9]. The system, however, does not function well if this assumption of independence isn't valid as shown in the KDD'99 data base, with complex dependencies on the features [10]. The results show that for the large datasets the Naïve Bayes model has decreased its accuracy.

3.3 Support Vector Machines (SVM)

Support Vector Machines (SVM) is a group of supervised learning methods with algorithms to analyze data and identify data classification patterns or regression analyzes. The SVM algorithm is based on statistical probabilistic theory of learning [11].

It is based on the concept of decision-making planes which separate objects with various classes of membership which define boundaries of decision. SVM is a classificatory of discrimination defined by a hyperplane splitting. SVMs use a kernel feature to map the workouts to a larger space to linearly classify intrusion.

SVMs are known for their capacity for generalization and are valuable especially when the attributes of the system are large, and the data points are small.

In IDS datasets, many functions in distinguishing data points to correct classes are redundant or less influential. The selection of features during SVM training should therefore be considered.

4. Proposed Intrusion Detection Algorithm

This section explains details of the Proposed Intrusion Detection System (IDS). Figure 2 below illustrates proposed IDS which are based on Support Vector Machine (SVM) learning classifier.

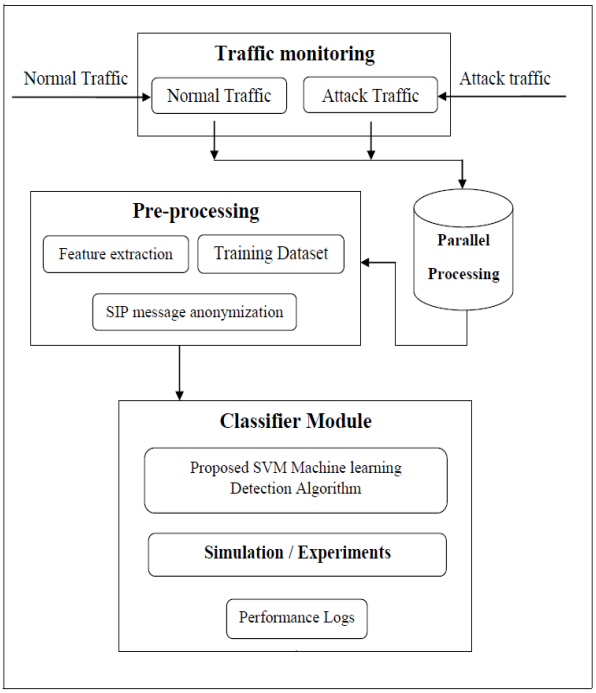

Figure 2. Proposed Intrusion Detection algorithm

Proposed Intrusion detection & preventions system methodology consists of the following modules: -

4.1 Traffic Monitoring

The first step is to monitor the incoming traffic; this paper considered two different traffic types; the first traffic type is the normal traffic type from legitimate clients and attack traffic type used for creating malicious traffic as shown in table 1. Below subsections explain details for each traffic type.

SIPp [12] is a free open-source traffic generator for the SIP protocol. It includes a few basic user agent scenarios (UAC and UAS) to establish and release multiple calls with the INVITE and REGISTER methods.

Table 1. Attack traffic generated by SIPp

Attack traffic type	Number of attack messages
Malicious / DoS flood attack	10000

4.2 Parallel Processing

The Parallel NIDPS was used for the proposed solution. Parallel NIDPS is a type of calculation where numerous NIDPS nodes simultaneously function according to the fact that large incoming data can be processed simultaneously because of creating small groups of information. This parallel processing is based on Apache Hadoop software [13].

This Hadoop collection may include a one primary parallel processing and several message handler nodes (secondary) as shown in Figure 3. The primary parallel processing node contains of a Job Tracker, Task Tracker, Name Node and Data Node. A secondary node acts as both a Data Node and Task Tracker.

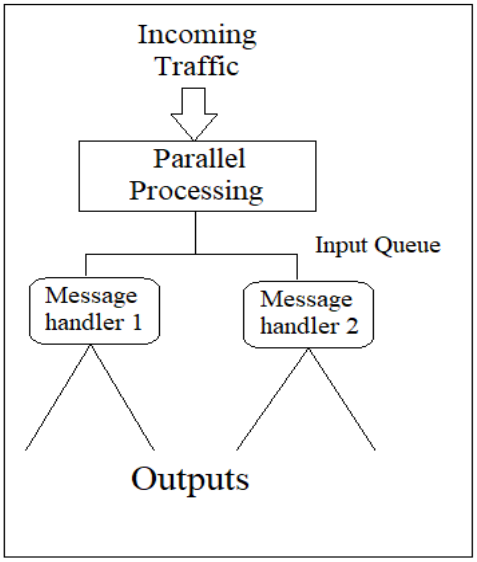

Figure 3. Parallel processing

4.3 Pre-Processing

Parallel Pre-processing does the following functions: -

Training Dataset

Datasets are valuable ways to carry out analytical assessments, to perform comparative study of different techniques and methods. However, the safety domain of the network, in particular network ID, lacks good quality data sets for analysis and testing of new algorithms and technology.

Many factors, including the rapid growth of this region, contribute to this. The highly evolving nature of network traffic and the large number of regular attack types have tried to produce them further. Details about the dataset are found in Table 2.

Table 2. Real-time world SIP training datasets from IP-PBX

Description	Number of messages
Total messages	34702
Total extracted training datasets	30000

Real-time training datasets are in PCAP format and python scapy library [14] is used to convert from "PCAP" to "txt" file.

Feature extraction

Feature extraction is essential for anonymization to extract features. For this purpose, SIP transactions and in some cases SIP dialogs shall reflect the features selected as shown in table 3 below.

Table 3. Selected feature extraction

Attribute	Description
METHOD CSeq Method	It shows the form and related transaction of the SIP message.
FROM, From TAG	It matches the logical originator of the application and its randomly picked TAG from which area of the FROM header
TO, TO TAG	It matches the rational receiver of the application and the TAG chosen randomly from the TO header area. A dialog between two UAs can be followed.
VIA Branch	A branch parameter which is unique in each transaction is given in the field in the SIP header
Call-ID	For each dialog it uses a powerful universal variable. The combination of a random string and the hostname/IP address of the softphone produces it. This role comes from the SIP header Call-ID field.

4.4 SIP message anonymization

It is not hard to collect a "true" SIP track from a SIP deployment. But, due to concerns about the privacy of the customers, the owners of infrastructures are reluctant to share this information. We have a deal to collect real SIP traffic with the local IP-PBX office that allows SIP tracks to be collected by mirroring the port before the IP-PBX server.
Mirror ports scan and collect all SIP messaging passing on the SIP servers, in conjunction with incoming and outgoing calls. To minimize privacy and security issues, the SIP messages are anonymized prior to stowing the captured SIP tracks in your local storage to conceal all confidential information that is accessible in a SIP message. Between May 2018 and February 2019, the network is up and running. There were captured over 30,000 SIP messages. To train the classifier, captured SIP messages are important.
The pipeline process is used to capture and anonymization SIP traffic. The monitoring process is performing the scanning process and saves SIP traffic on the IP-PBX server in a momentary safeguard on a single server. A script, containing anonymization codes, is used for anonymization purposes. It has a time-controlled trigger activated.

4.5 Classifier Module

SIP message traffic analyzer & Machine learning detection algorithms, the first SIP flooding examines each SIP message to determine if it is part of the official grammar language of the protocol for SIP (IETF, 2018). With the formal grammar description of SIP messages (the syntax of all SIP messages) any standard tools available under Unix can be applied to the Traffic Analyzer.
In a nutshell, a multistage classification is being carried out in this research: the first part consists of a quite straightaway SIP flooding analyzer that checks if the message is in accordance with RFC3261(IETF, 2018) SIP grammar.
The second part is the same. The specific functions for detecting SIP flood attacks are shown in Table 4.

Table 4. SIP flooding detection features

#	Features
1	Number of INVITEs & its requests per second
2	Number of REGISTERs & its packets
3	Number of 2XX & its number of requests
4	Number of transactions & its number of packets
5	Number of branches & its number of packets
6	Number of 4XX & its number of responses
7	Source IP address
8	Destination IP address
9	From Caller URI address
10	To Callee URI address

Analysis considered each feature's average and variance. Throughout the execution time, the performance of the respective function can be compared to its average under normal conditions. If the rate in normal situations is less than the variance, the flow is marked as normal and, otherwise, an alarm is generated. Proposed algorithm is shown in Figure 4.

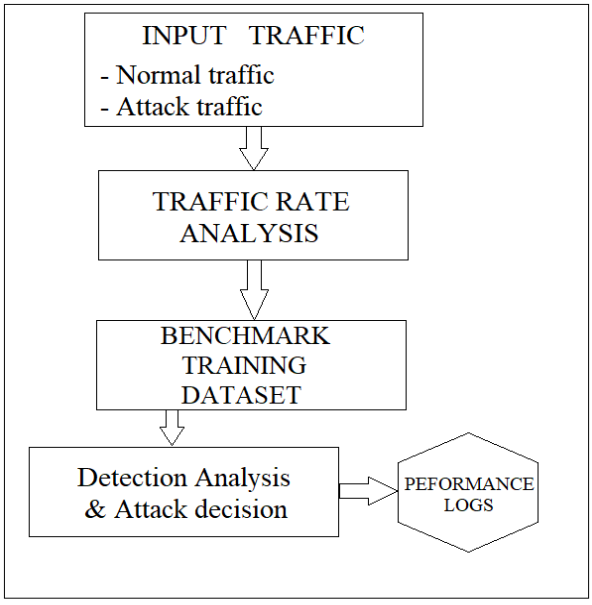

Figure 4. Classifier Module

Proposed algorithm performs the following functions: -

Input Traffic

• It continuously checks attack traffic & Normal Traffic from pre-processing module

• Check input traffic using below, this process continuously collects and tests network traffic and packet transfer rate traffic data from the machine learning.

Traffic rate analysis

The following the dynamic thresholds for rates of traffic use traffic-based statistics and their estimation methods.
1.Current Edge: Present levels of network traffic.
2.Normal Edge: Network traffic average rate for the period chosen.
3.Suspect Edge: Middle point between the network traffic rate limit and average values for the chosen duration, for example., value (current edge + normal edge)/2.
4.Attack Edge: Total network traffic within the range examined.

Benchmark input traffic

Benchmark received input traffic to Training dataset.
Detection analysis & Attack decision
1.SIP message flooding analyzer monitors whether the message has been complied with the standard of the SIP protocol as in IETF standard [15].

2. Select required features for detection using filter method as per below: -
- Number of INVITEs & its requests per second
- Number of REGISTERs & its packets
- Number of 2XX & its number of requests
- Quantity of transactions & its packets
- Quantity of branches & its packets
- Quantity of 4XX & its number of responses

3. Check the messages that has same origination & destination IP address, To URI, and Call-ID information.

In this study, a basic decision technique is used to check the pre-processed traffic and measure the mean of each function as a baseline reference under normal conditions and compare its difference with the runtime value. A simple decision mechanism is deployed to make decisions about the nature of the traffic (Normal or malicious).

A block diagram of the proposed SVM attack detection algorithm is illustrated in Figure 5.

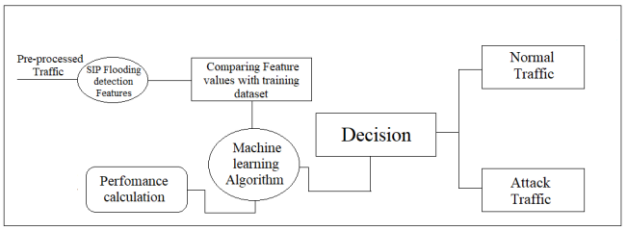

Figure 5. Proposed SVM Attack Detection algorithm

4.6 Intrusion Detection Performance indicators

The performance report log module is designed to collect KPIs from SIP customers and the IP-PBX server. During an experiment, every customer produces a file in the "CSV" format. Likewise, hardware resources statistics are measured and logged in real time on the server computer.
If an experiment is complete, the report generation module communicates the attacking node to SIP clients, SUT server, and produces a report that is used for performance metrics.
Several specific metrics are used in classification models to test machine learning algorithms. To track the performance of the established model several performance assessments functions have been tested. The measures are referred to as the uncertainty matrix.

A confusion matrix has two measuring factors commonly used for assessing classification model performance which are True Positive (TP) and False Negatives (FN).

True Positive (TP)

True positive (TP) measures the ratio between the accurate percentage of the attacks that we predicted and the total number of attacks that happened. Since all intrusions are intercepted, the TPR is 100% which is uncommon for IDS. TPR is also known as the sensitivity or detection rate.

False Negative (FN)

False Negative (FN) is when a detector fails to identify an anomaly as abnormal traffic.

Accuracy in attack detection

The accuracy of the flooding attack detection method is the ratio of instances correctly found to the total number of instances. The lower the number of false positives is, the better off it is for the system. The costs caused by false positives cause loss to the system and services provided.
A system that is categorized as a "false negative" will affect legitimate users but will not be able to successfully identify threats.
The accuracy or true positive (TP) is the number of correct alarms divided by the correct number of alarms plus false alarms, as shown in the following equation 1.

$$\text{Accuracy} = \frac{TP}{TP + FN} \quad \text{(Equation 1)}$$

Where: -
TP, FN is number of correct alarm and false alarm, respectively.

4.7 Experiment modules

Simulation has different modules as listed in table 5.

Table 5. Experiment 1 modules

#	Component	Description
1	SIP Server	Asterisk IP-PBX server
2	SIP clients	Softphones (Clients)
3	Attack call generation	SIPp is a traffic generator tool that Was explicitly developed for the purpose of research. SIPp is capable of simulation, and both Signaling and media traffic can also be generated by SIP clients.
4	Report module	This facilitates Statistics reports needed for analysis purpose.

4.8 Simulation: Experiment 1

Experiment 1 is the proposed SVM algorithm with proposed training dataset.
Experiment 1 is to evaluate proposed SVM architecture using proposed real-world training datasets from the IP-PBX. The efficiency of the proposed architecture is determined by its efficacy, which is defined by the accuracy of the classification, and its performance, which is the time taken for classification.
Achieved accuracy by SVM detection algorithm using proposed training dataset from IP-PBX is shown in Table 6.

Table 6. SIP DoS attack traffic used in experiment 1

Synthetic test scenarios	Number of attack messages
INVITE & REGISTER Attack messages	10000

The achieved accuracy by SVM detection algorithm using proposed training dataset from IP-PBX is shown in Table 7.

Table 7. DoS detection performance using proposed SVM.

Traffic statistics	Number of messages
Total (INVITE & REGISTER) messages	10,000
True positive ("malicious" cases acknowledged as "malicious")	9913
False positive ("good" cases acknowledged as "bad")	0
True negative ("good" cases acknowledged as "good")	0
False negative ("bad" message acknowledged as "good")	87
Accuracy	99.13%

4.9 Comparison of related training datasets

Research is based on real SIP traffic captured on real network; other researchers are using different non-real data networks. For cyber security, machine learning studies generally use packet-level data and public data [16].

Evaluation datasets play a crucial role in validating any IDS strategy, helping us to determine the potential of the proposed method to identify disruptive behavior.

Owing to privacy concerns, datasets used for network packet analysis in commercial products are not readily accessible. Nonetheless, some publicly accessible datasets such as DARPA, KDD CUP, NSL-KDD are commonly used as references to each other. However, the mostly used data in the cyber-security field is public data sets. They are generally KDD-CUP 99 and NSL-KDD data sets [16].

DARPA 1998 data set was created in Massachusetts Institute of Technology Lincoln Laboratory for testing of the IDS/IPS [17]. General approaches in machine learning based intrusion detection systems can be classified into three groups: anomaly-based, signature-based and hybrid. While some researchers focus on only the attack detection (Normal or Attack), others focus on the attack classification (Classification of each specific type of attack).

The authors [18] propose a hybrid SVM classifier for combining one-class SVM (unsupervised) and Soft-Margin SVM (supervised). They work on DARPA 1999 data set and use filter-based feature selection algorithm. Their research provides 90.90% accuracy and on DARPA 99 data set accuracy rate of about 92.19%.

The authors [19] test another SVM model. They use the NSL-KDD data set. Their approach is attack detection and they use filter-based feature selection. They achieve 82.68% accuracy rate. However, the accuracy of each attack class is not provided.

Another research paper by the authors [20] use Genetic Algorithm-Logistic Regression (GA-LR) for feature selection and C4.5 decision tree for multiclass classification on UNSW-NB15 and KDD-CUP 99 data set. Using only 20 features on UNSW-NB15 data set, they achieve 81.42% accuracy. Also, they agree that UNSW-NB15 data set is a more complex data set than KDD-CUP 99 data set.

Authors [21] used CIC-IDS2017 dataset which is an intrusion detection evaluation dataset from Canadian Institute for Cyber security team. In this research author used to benchmark NSL-KDD & CICIDS datasets.

NSL-KDD Training dataset models

NSL-KDD is a data set suggested to solve some of the inherent problems of the KDD'99 data set which are mentioned by authors [22] Although, this new version of the KDD data set still suffers from some of the problems discussed by McHugh [22] , Because of the lack of public data sets for network based IDSs, we assume that it can still be used as an efficient benchmark data set to help researchers compare various methods of intrusion detection and may not be a perfect representation of actual real networks.

In addition, the number of train and test sets of NSL-KDD records is appropriate. This advantage makes it economical without arbitrarily choosing a small section to run the experiments on the full spectrum. The outcomes of the assessment of different research papers would be reliable and compatible.

As shown in Table 8, NSL-KDD train dataset consists of 125,973 records and extracted test dataset contains 30,000 records.

Table 8. NSL-KDD DoS training datasets

Description	Number of messages
Total found Messages	125,973
Total extracted DoS attacks	30,000

CICIDS Training dataset models

CIC DoS IDS 2017 dataset is an intrusion detection evaluation dataset from Canadian Institute for Cyber security team [23]. The dataset contains both normal and common attacks which are most up to date as of 2017.

The CICIDS attack dataset is stored as "csv" file format and we convert it to "txt" file. CICIDS is fully labelled dataset. The dataset also includes the results of traffic analysis based on the timestamp, source and destination IPs, source and destination ports, protocols, and attack [24].

As shown in Table 9 below, 35000 records instances have been extracted and provided as training data in Table 9 after the import of the CICIDS data collection into the SQL server for 2019. Drawing on a detailed review of the data from CICDIDS the distribution of various forms of attacks was saved. Nearly 86% of the data obtained was attacked with DOS.

Table 9. CICIDS DoS training datasets

Description	Number of messages
Total found Messages	35000
Total extracted DoS attacks	30,000

Below Table 10 summarizes the characteristics of each of the datasets used in this research.

Table 10. Comparison of proposed dataset, CICIDS & NSL-KDD datasets

#	Training dataset name	Realistic data	Full packet captured	Extracted dataset for training	Year
1	Proposed real world dataset from IP-PBX	YES	YES	30000	2019
2	CICIDS	YES	YES	30000	2017
3	NSL_KDD	YES	YES	30000	2009

4.10 Simulation: Experiment 2

Experiment 2 is to illustrate and benchmark different Algorithms using different training datasets.

The next step is the implementation of the test process after the creation of the training models. 30000 DoS attack messages have been extracted from each dataset for a reasonable trial process. Extracted test information includes all sorts of CICIDS attacks.

Training time of different machine classifiers

When comparing the performance of the algorithms, the time needed for the training the algorithm of the given dataset is also considered. From table 11 SVM takes less time in training the model.

Table 11. Comparison of training dataset build time for different machine learning & different training datasets.

Machine learning classifier	Detection Time (Minutes)		
	Proposed training dataset	CCIDS dataset	NSL-KDD dataset
Proposed SVM	1.3	7.3	11.7
Decision tree	7.5	13.4	18.4
Naïve Bayes	18.3	25.3	31.8

Clearly, the proposed SVM took more than six times less than the time taken in training building by decision tree machine learning. In Naïve bayes takes more than 15 times as long as the SVM, this also means that proposed SVM requires less computational power to compared algorithms as explained in table 11.

Detection results using different machine learning algorithms.

A classification algorithm can use many evaluation metrics. The confusion matrixes were created for each classifier in this research. It includes information on existing and predicted output classes. Table 12 presents the TP rate values of the selected datasets in the experiment.

Table 12. Comparison of TP rate using proposed dataset CICIDS & NSL-KDD datasets for different machine learning classifiers.

Machine learning classifier	True positive (TP) Detection accuracy		
	Proposed training Dataset	CICIDS dataset	NSL-KDD dataset
	TP rate	TP rate	TP rate
Proposed SVM	99.13%	76.47%	65.17%
Decision tree	93.28%	63.71%	51.96%
Naïve Bayes	86.41%	41.58%	38.26%

It can be determined that the projected dataset for SVM algorithm achieved highest TP rate of 99.13% while decision tree and Naïve Bayes has 93.28% & 86.41% of TP rate, respectively. For CICIDS dataset, SVM algorithm achieved highest TP rate of 76.47% while decision tree and Naïve Bayes has 63.71% & 41.58% of TP rate, respectively.

Finally using NSL-KDD training dataset, SVM achieved 65.17%, while decision tree and Naïve Bayes has 51.96% & 38.26% of TP rate respectively; this is also shown in figure 6.

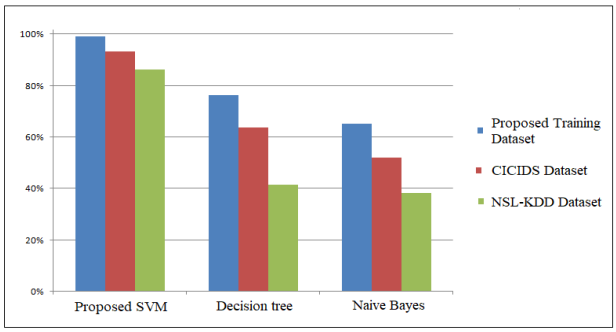

Figure 6. TP rate comparison for proposed dataset CICIDS & NSL-KDD datasets

TP rate and precision values are important attributes that define for a popular intrusion detection system, and from another viewpoint, the most serious performance parameters are false negative ("bad" message identified as "good"). The research papers of IDS are aimed to increase the TP rate and as well to decrease FN parameter as much as possible.

It is very important to show False negatives (FN), That means when an anomaly is not recognized and treated as ordinary traffic by a detector. Table 13 & figure 7 presents FN messages of the selected classifiers in the experiments.

Table 13. Comparison of FN messages using proposed dataset CICIDS & NSL-KDD datasets

Machine learning classifier	FN messages		
	Proposed training dataset	CCIDS dataset	NSL-KDD dataset
Proposed SVM	87	2353	3483
Decision tree	672	3629	4804
Naïve Bayes	1359	5842	6174

As illustrated in Table 13 the FN performance parameters, it can be concluded that the proposed dataset for SVM classifier achieved the lowest FN rates of 87 messages and 2353 messages & 3483 messages using proposed dataset and CICIDS dataset & NSL-KDD dataset respectively as illustrated in figure 7.

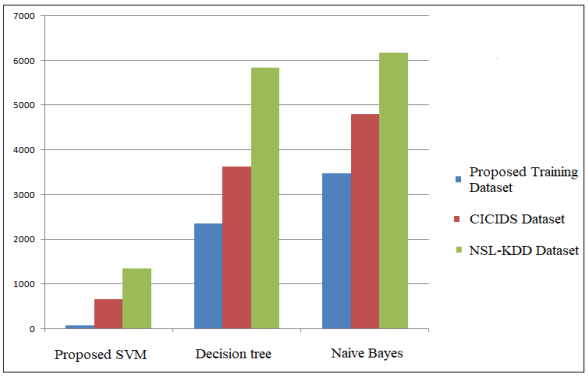

Figure 7. Comparison of FN messages for proposed dataset CICIDS & NSL-KDD datasets

In comparing the performance of the algorithms, the time required by algorithms to classify the attack is considered. This evaluation is conducted to show the effect of the machine learning model used. In the detection phase, this research compares the time consumed by different machine learning algorithms with different training datasets, as shown in table 14 and figure 8. The proposed SVM algorithm with proposed training dataset has the smallest detection time.

TP rate and detection time values are important attributes that define for a popular intrusion detection system, and from another viewpoint.

Table 14. Comparison of detection time for different machine learning & different training datasets.

Machine learning classifier	Detection time		
	Proposed training dataset	CCIDS dataset	NSL-KDD dataset
Proposed SVM	2.9	9.5	12.3
Decision tree	13.6	17.8	21.7
Naïve Bayes	26.2	32.5	42.6

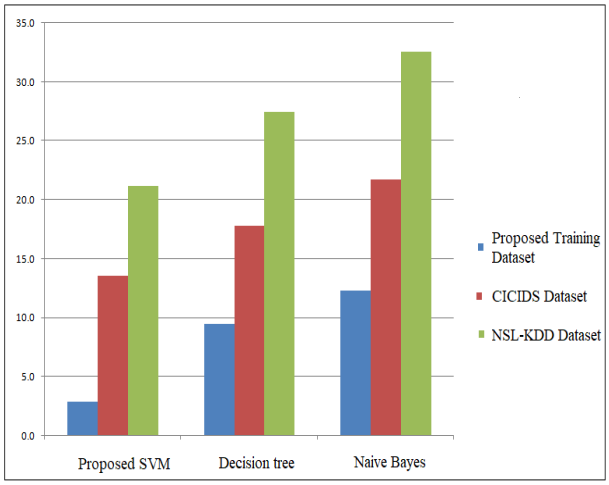

Figure 8. Comparison of Detection Time for proposed dataset CICIDS & NSL-KDD datasets

5. Conclusions

This paper examined current security status against SIP based IP-PBX. Research demonstrated the most common SIP based DoS attacks and their classifications, and then developed simulation architecture and experiment setup to analyze DoS attacks and their impact on SIP based IP-PBX servers.

This paper proposed a novel algorithm to detect and measure DoS attacks on SIP based IP-PBX. A novel algorithm design to mitigate and prevent Denial of Service (DoS) attacks on SIP based IP-PBX servers.

The paper examined that the proposed SVM algorithm can categorize the attack types with highest accuracy of 99.13%. As mentioned earlier, IDS systems have used old datasets. KDDcup99 and NSL-KDD are almost 15 years older, meaning that such training datasets cannot be used currently as network packet attributes have changed and were upgraded.

In this paper, IP-PBX real-time traffic datasets are used and as a comparison CICIDS 2017 dataset is used which has 30000 DoS attributes. The size and entry of the dataset are 20 times more than KDDcup99. Moreover, using our proposed real-time datasets shows highest detection accuracy. The proposed SVM algorithm provides a very good accuracy compared with other models with the same simulation dataset. For all the research and experimentation of NIDS with machine learning algorithm approach, these datasets are used for SVM and hence we compared our result with these available research papers.

Several experiments were conducted and tested in this research to evaluate the efficiency and performance of the following machine learning classifiers: SVM, Decision Tree and Naive Bayes. All the tests were based on the CICIDS intrusion detection dataset and our proposed real network datasets. The testing phase is based on 30,000 random instances of IP-PBX real-traffic training records and 10,000 messages of actual flooding attacks.

Several performance metrics are being computed (accuracy rate, False negatives, training build time and detection time). Other than the proposed SVM classifier, all other experiments have shown that there is no single machine learning algorithm capable of handling all types of attacks successfully.

The proposed detection methodology was measured in terms of false negatives and detection accuracy. False negativity means when a detector cannot recognize and categorize an anomaly as normal, while detection accuracy is an ability of identifier to detect an attack with higher accuracy value for getting better detection results.

Proposed real-time training dataset for SVM algorithm achieved highest detection rate of 99.13% while decision tree and Naïve Bayes has 93.28% & 86.41% of attack detection rate, respectively.

For CICIDS dataset, SVM algorithm achieved highest detection rate of 76.47% while decision tree and Naïve Bayes has 63.71% & 41.58% of detection rate, respectively. Using NSL-KDD training dataset, SVM achieved 65.17%, while decision tree and Naïve Bayes has 51.96% & 38.26% of detection rate, respectively.

On the other hand, the execution time taken by the algorithms to classify the attack is very important. The execution time taken by the algorithms to classify the attack is very important. SVM gives less time (2.9 minutes) for detecting attacks while decision tree and naïve Bayes gives 13.6 minutes 26.2 minutes, respectively.

Proposed SVM algorithm the lowest false negative value obtained (87 messages) while decision table (rules base classifiers) and Naïve Bayes achieved false negative messages of 672 and 1359, respectively.

This proposed model could be implemented for the detection of any DoS attack and unknown data type. One of the most important concerns to consider is that to make sure the data is clean. From the results and analysis, proposed SVM algorithm with NIDS is a fair consideration for the enhancement of the improved accuracy of intrusion detection in the network with high performance metrics.

There are not enough (existing) datasets to properly interpret the new attacks, and therefore the need for new (up to date) datasets arises. However, building new datasets effectively depends upon the experienced knowledge of the mapper who labels the data which is expensive and time-consuming. To fair represent the research questions, we recommend having data more and less complete, and storing it with less redundancy. It has been known that, in some cases, organized datasets along with deep learning provide a solution to a problem.

It is recommended to retest performance of machine learning algorithms proposed here on actual operator network by increasing the dataset size.

References

[1] Husak, M., Komarkova, J., Bou-Harb, E. and Celeda, P. "Survey of Attack Projection, Prediction, and Forecasting in Cyber Security". IEEE Communications Surveys & Tutorials, 21(1), pp.640-660, 2019.

[2] G. Vasconcelos, R. S. Miani, V. C. Guizilini and J. R. Souza. "Evaluation of DoS attacks on Commercial Wi-Fi-Based UAVs", International Journal of Communication Networks and Information Security (IJCNIS), Vol. 13, No. 1, pp. 212-223,2021.

[3] Spafford, E. and Zamboni, D. "Intrusion detection using autonomous agents. Computer Networks", 34(4), pp.547-570, 2000.

[4] Khraisat, A., Gondal, I., & Vamplew, P. "An Anomaly Intrusion Detection System Using C5 Decision Tree Classifier", PAKDD, 2018.

[5] Butun, I., Morgera, S. and Sankar, R. "A Survey of Intrusion Detection Systems in Wireless Sensor Networks". IEEE Communications Surveys & Tutorials, 16(1), pp.266-282,2014.

[6] Sumeet, D and Xian D. "Data Mining and Machine Learning in Cybersecurity", Auerbach Publications, 2011.

[7] Sharma, A., & Parihar, P. "An Effective DoS Prevention System to Analysis and Prediction of Network Traffic Using Support Vector Machine Learning", 2013.

[8] Rutkowski, L., Jaworski, M., Pietruczuk, L. and Duda, P. "Decision Trees for Mining Data Streams Based on the Gaussian Approximation", IEEE Transactions on Knowledge and Data Engineering, 26(1), pp.108-119,2014.

[9] X. Yang and Y. L. Tian. "EigenJoints-based action recognition using Naïve-Bayes-Nearest-Neighbor," IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, rovidence, RI, USA, pp. 14-19,2012.

[10] Koc, L., Mazzuchi, T. and Sarkani, S. "A network intrusion detection system based on a Hidden Naïve Bayes multiclass classifier", Expert Systems with Applications, 39(18), pp.13492-13500,2012.

[11] Vapnik, V. "The nature of statistical learning theory", New York: Springer, 2010.

[12] SIPp, version 3.4. SIPp. http://sipp.sourceforge.net, 2018.

[13] Hadoop, parallel processing software. Hadoop. https://hadoop.apache.org, 2019

[14] Python scapy library. Python. https://pypi.org/project/scapy, 2019.

[15] IETF Network Working Group. SIP: Session Initiation Protocol. IETF. http://www. ietf.org/rfc/rfc3261.txt, 2018

[16] Buczak, A. and Guven, E. "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection", IEEE Communications Surveys & Tutorials, 18(2), pp.1153-1176,2016.

[17] Nawir, M., Amir, A., Lynn, O., Yaakob, N. and Badlishah Ahmad, R. "Performances of Machine Learning Algorithms for Binary Classification of Network Anomaly Detection System", Journal of Physics: Conference Series, 1018, p.01,2018.

[18] Shon, T. and Moon, J. "A hybrid machine learning approach to network anomaly detection", Information Sciences, 177(18), pp.3799- 3821,2007.

[19] Pervez, M. & Farid,D. "Feature selection and intrusion classification in NSL-KDD cup 99 datasets employing SVMs", The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014), pp. 1-6, 2014.

[20] Khammassi, C. and Krichen, S. "A GA-LR wrapper approach for feature selection in network intrusion detection", Computers & Security, pp.255-277,2017.

[21] Mohammed, A, Najla, B. "A detailed analysis of new intrusion detection dataset", Journal of Theoretical and Applied Information Technology, Vol.97. No 17, 2019.

[22] M. Tavallaee, E. Bagheri, W. Lu and A. A. Ghorbani. "A detailed analysis of the KDD CUP 99 data set", 2019 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada, pp. 1-6, 2019.

[23] CIC DoS dataset. Canadian Institute for Cyber security, 2019.

[24] Sharafaldin, I., Lashkari, A.H., & Ghorbani, A. "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization", ICISSP, 2018.