IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 27, Issue 6, Ser. 1 (Nov. – Dec. 2025), PP 79-87
www.iosrjournals.org

Design of IDS Framework to Detect TCP-SYN Flood Attack in Cloud Environment Using ML Approach

Khwahish Indoria
Department of Computer Science and Engineering

Dr. Nimish Kumar
(Professor and Head)
Department of Computer Science and Engineering

Abstract
A network traffic monitoring system known as an Intrusion Detection System (IDS) is a system that analyses network traffic in order to detect suspicious actions and sends out notifications whenever such an activity is identified. The program is a piece of software that performs a scan of a system or network to look for malicious behaviour or violations of regulations. In contrast to signature-based intrusion detection systems (IDS), the machine learning-based technique has a more generalised property. This is because the models used in the machine learning method may be developed according to the applications and hardware configurations. Through the use of an anomaly-based detection engine, it is possible to identify deviations from the typical state of the system, which may be the result of an attack on the system.
Keywords: *IDS Detect, TCP-SYN, Flood, Attack, Cloud Environment, Ml, Approach*

I.Introduction

Intrusion Detection System (IDS)

A network traffic monitoring system known as an Intrusion Detection System (IDS) is a system that analyses network traffic in order to detect suspicious actions and sends out notifications whenever such an activity is identified.

The program is a piece of software that performs a scan of a system or network to look for malicious behaviour or violations of regulations.

Any harmful activity or violation is often reported to the administrator or gathered centrally using a Security Information and Event Management (SIEM) system. Both of these methods are considered to be standard practice (Akana Chandra, 2021).

A security information and event management (SIEM) system employs alarm filtering algorithms to discern between malicious activity and false alarms, and it combines outputs from many sources. Additionally, placing false alerts in the intrusion detection system (IDS) and monitoring the networks for potentially harmful behavior

Normal Connection of Transmission Control Protocol (TCP)

In a network, the Transmission Control Protocol (TCP) is a connection-oriented communication protocol that makes it easier for computer devices to communicate with one another and share messages.

By using the Internet Protocol (IP), it is a component of the transport layer protocol that is responsible for establishing and sustaining connections between hosts in computing networks. Packets are sent from the source to the destination by the computer equipment in order to facilitate the exchange of messages across a network and to guarantee the uninterrupted delivery of data and messages (Chaudhary,2014).

A visual representation of the construction of a TCP segment. When communicating with one another, TCP hosts are required to create a connection-oriented session. Establishing the connection is accomplished by the use of the three-way handshake method, which serves to synchronise both the sender and the receiver of a network. This enables both the sender and the receiver to reach a consensus on the original sequence numbers.

Additionally, this process assures that both the transmitter and the receiver are prepared to transfer data and to understand that the sender or recipient is accessible to speak with one another (D'hooge, 2020).

Abnormal Connection of Transmission Control Protocol (TCP)

In a network, an abnormal connection is said to have occurred when the target server or system in the network got a greater number of SYN packets than usual. Figure 1.5 is an illustration of the anomalous connection

that TCP has. The TCP SYN assault is a sort of distributed denial of service attack, which is also known as a half-open attack.
During the half-open attack, the TCP connection requests are exploited in order to drain resources on the targeted server. This is done quicker than the machine that is being targeted.
It may render them unresponsive by sending initial connection request packets over and over again without taking the time to react to the acknowledgements that correspond to them.
Cybercriminals encourage an increase in the number of open TCP connections, which causes the server's resources to become essentially crowded out by legitimate traffic. As a result, it would be impossible to open new legitimate connection (Donadio, 2012) .

Detection Methods of IDS

1. Signature based Method
A signature-based intrusion detection system (IDS) is able to identify instances of assaults by analysing certain patterns in network traffic, such as the number of bytes, the number of ones, or the number of zeros. Additionally, it is able to identify dangerous software by analysing the sequence of instructions that are previously known to be bad and that are used by the infection. Signatures are the patterns that are identified by the intrusion detection system (IDS).
The signature-based intrusion detection system (IDS) is able to identify assaults with ease if the pattern (signature) of the attack already exists in the system. However, it is particularly challenging to detect new malware attacks since their pattern (signature) is unknown (Eddy, 2007).

2. Anomaly based Method
Anomaly-based intrusion detection systems have been implemented in order to identify unknown malware assaults in light of the constant development of new malware. When an anomaly-based intrusion detection system (IDS) is implemented, a machine learning technique is used to generate a trustworthy activity model. Any new information that is received is compared to this model, and if it is not included in the model, it is deemed suspicious.
In contrast to signature-based intrusion detection systems (IDS), the machine learning-based technique has a more generalised property (Mirjalili, 2018). This is because the models used in the machine learning method may be developed according to the applications and hardware configurations. Through the use of an anomaly-based detection engine, it is possible to identify deviations from the typical state of the system, which may be the result of an attack on the system.

II. Review Of Literature

Deepti Lamba et al., (2021) have conducted an analysis of a pattern that would make it possible to mine valuable medical information from a community cloud that has connected a variety of health organisations. Cloud computing allows for the execution of classification algorithms in parallel over several processors. Some examples of these algorithms are Decision Tree, k-NN classifier, and Naive Bayesian classifier (Deepti Lamba et al., 2021). When it comes to the prediction of therapeutic effects, it is generally accepted that ensemble approaches such as bagging and boosting have been made possible to improve the accuracy of each and every classifier.

Ladislav Huraj. et al., (2019) have observed that cloud services are characterised by their flexibility, which is one of the most significant benefits for accommodating variations in data demand. There is a possibility that the use of data may result in ongoing changes to the uptime and the downtime of data transmission (Ladislav Huraj. et al., 2019). There is a possibility that the number of distributed denial of service assaults (DDOS) may quickly rise if they are not correctly addressed at a given point in time. During the course of the research, an experimentation that is analogous to a real-time cloud environment is carried out, and an attack scenario is conceived up and evaluated with the help of the newly acquired dataset in IDS.

Alka Chaudhary et al., (2014) have proposed that the contemporary day networking configurations are dependent on mobile ad-hoc networks, in which intrusion detection is built to prevent needless data breaches. This is due to the fact that data theft and data breaches have the potential to inflict serious harm to the mobile network system. Within the scope of the present investigation, a method that is based on fuzzy clustering is created in order to identify intrusions in MANET. The effectiveness of the proposed method is evaluated by contrasting its performance on the real-time dataset with that of the iris dataset in order to validate its effectiveness. This technique is implemented in a variety of datasets as a cloud storage system that is both distributed and capable of collaborative computation (Alka Chaudhary et al., 2014).

Mohd Naved Ul Haq and Narender Kumar (2021) have conducted an analysis of the effects of the security risk and offered a strategy to address the authentication and storage level threats that are associated with cloud computing environments. It has been suggested that a data classification approach that is based on the

principle of data secrecy might be used to achieve data categorisation (Mohd Naved Ul Haq and Narender Kumar 2021). Following this, an efficient security system has been implemented, which includes encryption, authentication, and authorisation, or all of these, in order to guarantee the confidentiality of data stored in the cloud.

Islam Md. M., et.al., (2019) In addition, it has been observed that cloud service providers have the ultimate duty for ensuring the safety of data while it is being exchanged or stored. Furthermore, the encryption mechanism ensures that only authorised individuals can access the data and that the data may be sent. It is necessary to have a suitable decryption key in order to get the data that was transacted (Islam Md. M., et.al., 2019). Nevertheless, there is a potential that data stored on a cloud server might be accessed by unauthorised parties.

Panigrahi R. et.al., (2020) have shown that there is a rise in the number of concerns about data privacy and security in cloud computing in general. A novel form of architecture that incorporates a generalised multiprotocol label switching is offered as a means of preventing problems of this nature in intrusion detection systems (IDS). The use of a contemporary architecture is suggested as a means of lowering the vulnerability of data management (Panigrahi R. et.al., 2020).

Shallal Qahtan M et.al., (2016) as a result of the fact that the number of distributed denials of service attacks in cloud computing environments has significantly grown, a number of proposals have been made for detecting them (Shallal Qahtan M et.al., 2016). The classification algorithm-based approach is the one that is used the most often among these strategies.

OBJECTIVES OF THE STUDY

1. To study on Intrusion Detection System (IDS)
2. To study on Normal Connection of Transmission Control Protocol (TCP)

III. Research Method

Feature Selection Methods and Classification Techniques

During this phase, a technique known as Machine Learning (ML) is used in order to construct an intrusion detection system (IDS) framework for the purpose of detecting distributed denial of service assaults (DDoS), which may include TCP-SYN floods. There is a possibility that a relevance analysis will be required to come before the categorisation model has to be implemented. During the classification process, this analysis not only makes an attempt to discover which characteristics are significantly useful, but it also makes certain that irrelevant characteristics are not taken into account (Buschendorf, 2017). In the process of creating phase-I, techniques for feature selection and reduction, such as LVQ and PCA, as well as classification algorithms, such as NB, SVM, and DT, are used. LVQ and PCA are two examples of these techniques. A representation of the architectural design that was carried out for the first phase may be seen in Figure 1.

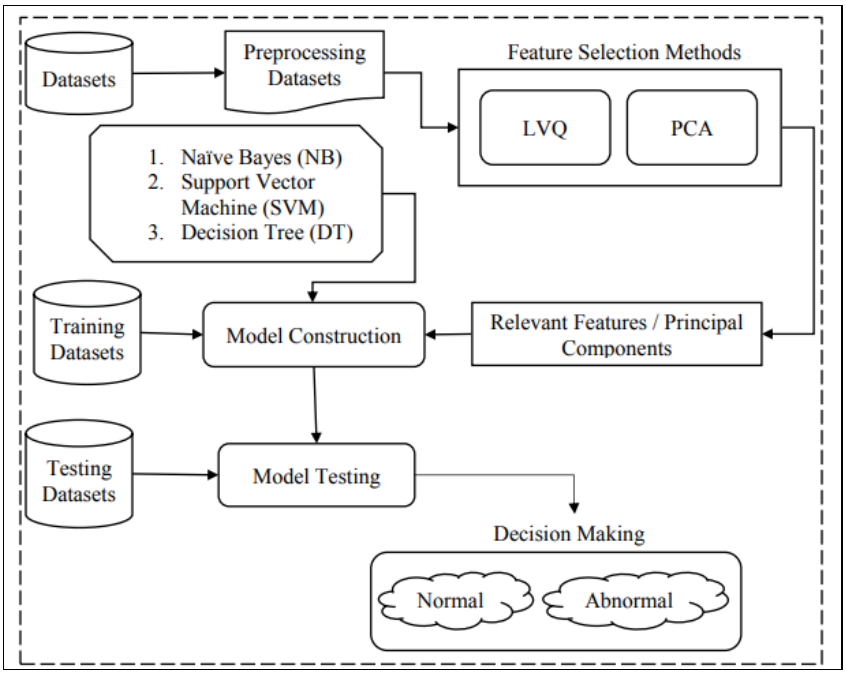

Figure 1 Architectural Design of IDS Framework to Detect TCP-SYN Flood Attack in Cloud Environment Using ML Approach

Feature Selection and Dimensionality Reduction

Preprocessing the data is one of the most important phases in machine learning, since it is responsible for putting the raw data into a format that is comprehensible and enforceable. This term refers to the process of preparing, cleaning, and organising the raw data that comes from the actual world, which may include contradictory and inconsistent information, in order to make it acceptable for the construction and training of machine learning models with increased efficiency. Both the quality of the data and the ability to get relevant insights from the data are improved as a result of this technology (Wu, H. 2018).

The pre-processing procedures include making changes to the data and arranging them in a reasonable manner. The pre-processing of data that is both complicated and high dimensional is accomplished via the use of feature selection or dimensionality reduction techniques. A strategy that is used to pick the significant characteristics that have a greater influence on the variable that is being forecasted is known as the feature selection approach. The process of selecting features will not be changed in any way, even if the inclusion and exclusion of data take place, and it will also remove characteristics that are much less discriminative and redundant (Doegar, E. A. 2018). The available features are separated into a subset of the characteristics that are the most important, and this subset is used in the process of implementing the machine learning technique. In the field of machine learning, feature selection is used for the reasons that are listed below.

- To increase the accuracy of the trained model because a few relevant features are better to train the model than a huge amount of irrelevant and redundant feature.
- To avoid overfitting and dimensionality issues.
- To increase the accuracy and the efficiency of the classifier.

The dimensionality of a dataset is an indicator of the number of variables or characteristics that are used to construct the dataset. When there are a greater number of characteristics included inside a dataset, the process of developing a model will therefore become more difficult. Therefore, in order to lessen the difficulty of this difficult endeavour, dimensionality reduction methods are used. Techniques that lower the number of input variables in a dataset are an example of what is meant by the term "dimensionality reduction." Not only does it improve data visualisation, but it also helps reduce the amount of time and space that is necessary for data processing tasks. In the current investigation, the retrieval of the relevant features is accomplished by the use of Learning Vector Quantisation (LVQ) and Principal Component Analysis (PCA). A statistical measure is used in the LVQ filter technique, which is a filter method that assigns a score to each unique characteristic (Herzberg, 2013). When the score is taken into consideration, the characteristics are either chosen to be retained in the dataset or eliminated from it. principal component analysis (PCA) is a technique for reducing dimensionality that is used to decrease dimensionality and locate data points that have the largest potential variance.

Learning Vector Quantization (LVQ)

An example of a supervised learning approach that is used in Artificial Neural Networks (ANN) is the Learning Vector Quantisation (LVQ) algorithm. Additionally, there are "n" number of input units and "m" number of output units included inside the design. LVQ is visualised via the use of a set of codebook vectors (Yin, 2017). A collection of integers that have the same input and output qualities as the training dataset is what is known as a codebook vector.

A predefined pool of codebook vectors, which were learnt from the training dataset, is used to represent the model via its representation. In spite of the fact that the vectors have the same appearance as the training examples, the learning mechanism has applied modifications to the values of each attribute. In order to build the LVQ network, codebook vectors are used as neurones, and the weight value for each attribute is applied to the corresponding attribute.

To locate the values that are closest to one another in the LVQ network, the K-Nearest Neighbours (KNN) algorithm is used. This allows for the selection of the available features to be made. Through the use of the KNN method, a new piece of data (x) is searched from the vectors in the code book for the K examples that are the most similar to it, and the output variable from each of the K instances is summarised (Shaikh, 2022).

It is referred to as the Best Matching Unit (BMU) when the K value is equal to one. This is done in order to identify the relevant characteristics. The code book vector and K instances are matched. A Euclidean Equation (1) is used to determine the distance between each codebook and the new piece of data. This distance is then used to select the relevant K occurrences in a training dataset (Valavan, 2024). The following is an example of how they are calculated.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2} \quad \text{…..(1)}$$

Learning the training data in LVQ is done in a random fashion, using the codebook vectors that have been created. It is decided at random whether there will be twenty, thirty, or forty codebook vectors. In the course of carrying

out this procedure, the input of training data and the selection of random codebook vectors are seen as being identical to the output of the class. In the event that the output of the codebook vector and the training instance are identical, the vector will be pushed closer to the training instance whenever this occurs. If they do not match, it is shifted farther away from both of them. Using Equation (2), the amount of movement of the vector is determined by the learning_rate parameter used in the equation (Gnana Singh, 2013).

$$earningrate = \alpha * (1 - (epoch/maxepoch)) \quad (2)$$

A specification of alpha and maxepoch is made at the beginning of the run, where learning_rate is the value that corresponds to the current epoch (ranging from 0 to maxepoch -1). For the purpose of this study, the parameters (α = 0.3, epoch = 0 and maxepoch = 200) have been settled upon. It is possible for the learning rate to have a high value during the first epoch, but by the end of the epoch, it may be reduced to a low value.
The majority of the modifications to the codebook vectors occur at the beginning of the process, whereas the changes that occur at the conclusion of the process are extremely tiny adjustments. Equation (3.3) is used in the event that the training input variable (x) from the codebook vector is relocated to a more convenient location by the classes. Equation (3.4) is used in the event that the training input variable (x) from the codebook vector is relocated to the classes.

$$x = x + learningrate * (t - x) \quad (3)$$
$$x = x - learningrate * (t - x) \quad (3)$$

Where t is a training input value.
The LVQ parameters used for the training process are described in Table 1.

Table 1 LVQ Parameters and Descriptions

LVQ Parameters	Description
x	The training vector x(x1,x2……xn)
wj	Weight vector for j^{th} output unit
t	The class for training vector x
cj	The class associated with the j^{th} output unit

IV. RESULT

Performance Evaluation Measures

According to Diro, A., et al. (2018), the prediction of class labels via the use of evaluation metrics is the means by which the evaluation of a good and accurate classifier is performed. Accuracy, precision, recall or sensitivity, specificity, and f-measure are only few of the many distinct types of evaluation metrics that are available. A number of terms, including True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN), are used in the process of calculating the evaluation. Table 2 displays the confusion matrix, which is created using the terms of assessment measures. The matrix is displayed in the table. Identifying the numerous classes that are used for the analysis of the data is the responsibility of the confusion matrix, which is also responsible for measuring the efficacy of the classifier (Jayapandian, 2020).

Table 2 Confusion Matrix

Actual Class / Predicted Class	Abnormal(1)	Normal(0)
Abnormal(1)	TP	FP
Normal(0)	FN	TN

- **True Positive (TP) :** If the predicted sample and actual sample are abnormal
- **False Negative (FP):** If the predicted sample is abnormal, but the actual sample stands normal, as well as type-I error.
- **True Negative (TN) :** If the predicted sample and actual sample are normal

➢ **False Positive (FN):** If the predicted sample is normal, but the actual sample stands abnormal, as well as type-II error.

Out of the total datasets, seventy-five percent are used for training purposes, while the remaining twenty-five percent are utilised for testing purposes. Assessment metrics are used in order to determine whether or not the model is successful. Tables 2 and 3, respectively, provide the confusion matrix of classification models that use LVQ and principal component analysis techniques (Jonathan, 2021). These examples are shown in the tables.

Table 3 Confusion Matrix Parameters of Classification Models using LVQ for Feature Selection

Datasets / Parameter	Public Dataset						Private Dataset		
	NSL-KDD			CIC-DDoS2019					
	NB	SVM	DT	NB	SVM	DT	NB	SVM	DT
TP	9211	9359	9418	10327	9424	11327	12563	11945	12784
FP	407	242	372	1286	1763	981	640	1258	419
TN	3197	3252	3339	1364	3209	2135	2964	5070	4681
FN	2727	2689	2413	3135	1716	1669	4895	2789	3178
Total Number of Records	15542			16112			21062		

Table 4 Confusion Matrix Parameters of Classification Models using PCA for Feature Reduction

Datasets / Parameter	**Public Dataset**						**Private Dataset**		
	NSL-KDD			**CIC-DDoS2019**					
	NB	**SVM**	**DT**	**NB**	**SVM**	**DT**	**NB**	**SVM**	**DT**
TP	9369	9507	9514	9856	9129	7856	11245	12371	12371
FP	342	281	297	2231	2165	967	1958	832	832
TN	3147	3752	3630	2607	3934	6707	6003	4911	5711
FN	2684	2002	2101	1418	884	582	1856	2948	2148
Total Number of Records	15542			16112			21062		

When referring to a classifier model, the term "accuracy" refers to the ability of the model to produce correct predictions (Sharma, 2021). The accuracy of the classifier is determined by the ratio that exists between the total percentage of TP and TN and the total percentage of TP, TN, FP, and FN. For the purpose of determining the level of accuracy that the model has, equation (4) is used.

$$Accuracy=(TP+ TN)/ (TP+TN + FP+FN) \quad (4)$$

An evaluation of the efficacy of the NB, SVM, and DT classification models is carried out by using one of the assessment metrics of accuracy, which are shown in Table 4.4. When comparing the results of classification models that use a combination of LVQ feature selection and PCA dimensionality reduction approaches, analyses are carried out on the comparative outcomes of these models. When applied to classification methods, it was found that the combination of LVQ and DT displayed greater performance in terms of accuracy (Kulwinder, 2023). This was discovered (Abraham, A., 2013). This was the situation with each and every one of the datasets that were

made available. Using principal component analysis (PCA) with support vector machine (SVM) for NSL-KDD and PCA with deep learning (DT) for both CIC-DDoS2019 and private datasets both provide attack detection results that are better to those obtained using other approaches.

Table 5 Accuracy Comparison of Classification Models

	Feature Selection/Reduction					
	LVQ			**PCA**		
Classification Models / Datasets	**NB**	**SVM**	**DT**	**NB**	**SVM**	**DT**
NSL-KDD	0.7984	0.8114	**0.8208**	0.8053	**0.8531**	0.8457
CIC- DDoS2019	0.7256	0.7841	**0.8355**	0.7735	0.8108	**0.9039**
Private	0.7372	0.8079	**0.8292**	0.8189	0.8205	**0.8585**

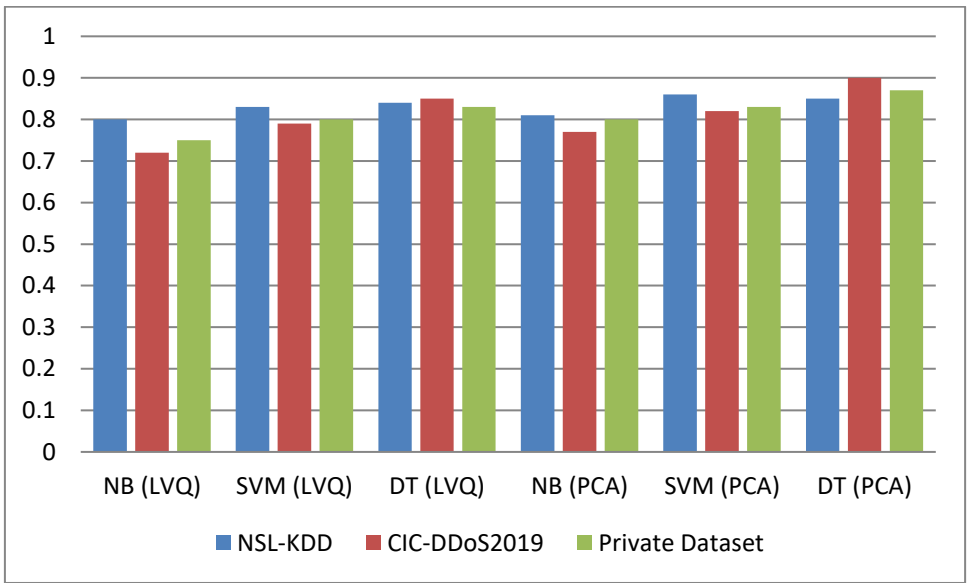

Figure 2 Accuracy Comparison of Classification Models

The performance of classification algorithms for attack detection is shown to be superior in PCA when compared with LVQ in terms of accuracy. This is seen when comparing the two methods. The detection of attacks may be accomplished using one of these two methods (Mehmood, 2012). A graphical representation of the comparison of the two approaches to accuracy may be seen in Figure 4.1. The positive predictive value is referred to as precision, which is a phrase used to characterise it. In addition, it is the proportion of TP to the sum of TP and FP both taken together (Maglaras, 2020). When doing the computation, the equation (5) is used in order to get the desired precision value.

$$\text{Precision} = TP/(TP+FP) \quad (5)$$

Table 6 presents the results of an analysis that measures the performance of categorisation models with regard to their level of accuracy. For the NSL-KDD dataset, the overall performance of LVQ with SVM is superior. While this is going on, LVQ with DT is giving superior results for both the CIS-DDoS2019 dataset and the private dataset. PCA with SVM produces higher TP for NSL-KDD, while PCA with DT produces better TP for CIC-DDoS2019. Both of these combinations are used. In the TP of principal component analysis, the combination of DT and SVM is superior for private datasets (Labraoui, 2023).

Table 6 Precision comparison of Classification Models

	Feature Selection/Reduction					
	LVQ			PCA		
Classification model Datasets	**NB**	**SVM**	**DT**	**NB**	**SVM**	**DT**
NSL-KDD	0.9577	**0.9748**	0.9620	0.9648	**0.9713**	0.9697
CIC-DDoS2019	0.8893	0.8424	**0.9203**	0.8154	0.8083	**0.8904**
Private	0.9515	0.9047	**0.9683**	0.8517	**0.9370**	**0.9370**

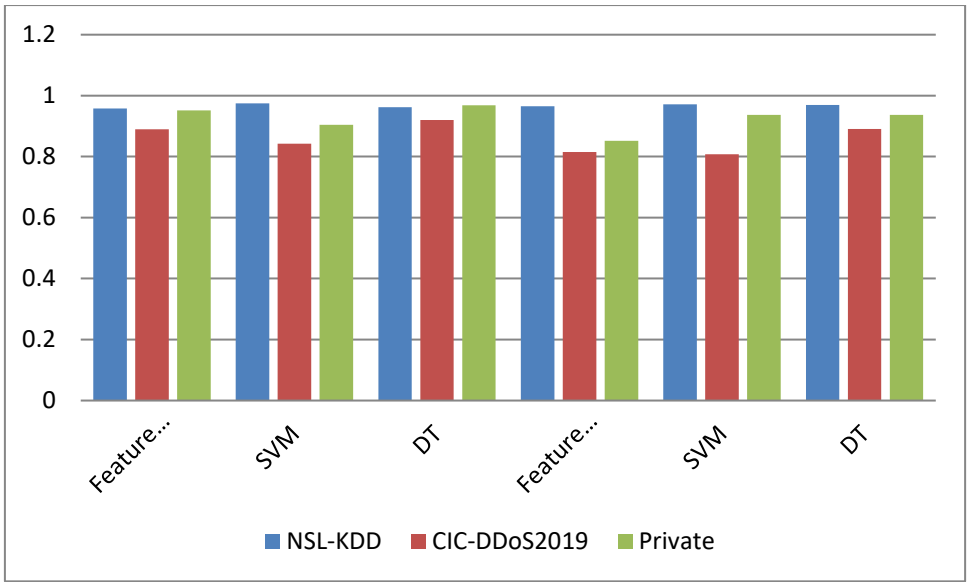

Figure 3 Precision Comparison of Classification Models

A graphical example of the comparison of accuracy may be seen in Figure 3. For the purpose of determining whether or not TP is accurate, recall, which is also known as sensitivity, is used. There is a statistic that is known as the TP Rate (TPR).

V. Conclusion

The cloud intrusion detection system (IDS) is built in three steps to recognise a TCP-SYN flood DDoS attack on three separate datasets: NSL-KDD and CIC-DDoS2019, two public datasets, and Real-Time Capture, a private dataset. Applying LVQ and PCA, two feature selection algorithms, to the supplied datasets is the initial stage. Classification models, such as NB, SVM, and DT, are built using the characteristics that are generated by these applications. In the NSL-KDD dataset, PCA with the SVM technique performed better. However, in the CIC-DDoS2019 dataset and the private dataset, PCA with the DT algorithm obtained the highest attack detection. Fuzzy logic rules are created in the second stage utilising the characteristics selected using principal component analysis as input. Repetitive fuzzy rules are removed by applying the MGWO algorithm. For the purpose of further improving the detection rate, we compare bagging approaches with simple classifiers. Improved detection performance was shown by the bagging techniques, which comprised bagged SVM on the NSL-KDD dataset and bagged DT on the CIC-DDoS2019 dataset as well as the private dataset.

References

[1] Akana Chandra, M. V. S., Satyanarayana, C., Divakar, C., & Katikireddy, P. S. (2021). Sentiment analysis using neural network and LSTM. *IOP Conference Series: Materials Science and Engineering, 1074*(1), 012007. https://www.researchgate.net/publication/349628900

[2] Chaudhary, A., & Tiwari, V. N. (2014). Analysis of fuzzy logic-based intrusion detection systems in mobile ad hoc networks. *Future Generation Computer Systems, 79,* 617–624. https://www.researchgate.net/publication/364198123

[3] D'hooge, L., Wauters, T., Volckaert, B., & De Turck, F. (2020). Inter-dataset generalization strength of supervised machine learning methods for intrusion detection. *Journal of Information Security and Applications, 54,* 135–143. https://doi.org/10.1016/j.jisa.2020.102563

[4] Donadio, P. (2012). Virtual intrusion detection systems in the cloud. *Bell Labs Technical Journal, 17,* 113–128. https://doi.org/10.1002/bltj.21506

[5] Eddy, W., & Chi, F. (2007). TCP SYN flooding attacks and common mitigations. *Journal of Information Technology,* 23–31. https://doi.org/10.1057/palgrave.jit.2000090

[6] Faris, H., Aljarah, I., Al-Betar, M. A., & Mirjalili, S. (2018). Grey wolf optimizer: A review of recent variants and applications. *Neural Computing and Applications, 30,* 413–435. https://doi.org/10.1007/s00521-018-3072-1

[7] Lamba, D., Hsu, W., & Alsadhan, M. (2021). Predictive analytics and machine learning for medical informatics: A survey of tasks and techniques. In *Handbook of Statistics* (Vol. 45, pp. 511–540). Elsevier. https://doi.org/10.1016/B978-0-12-821777-1.00023-9

[8] Huraj, L., & Simon, M. (2019). Realtime attack environment for DDoS experimentation. In *2019 IEEE 15th International Scientific Conference on Informatics (INFORMATICS)* (pp. 101–105). IEEE. https://www.researchgate.net/publication/342255349

[9] Haq, M. N. U., & Kumar, N. (2021). A novel data classification-based scheme for cloud data security using various cryptographic algorithms. *International Review of Applied Sciences and Engineering, 13*(2), 107–116. https://doi.org/10.1556/1848.2021.00317

[10] Islam, M. M., Hasan, M. Z., & Shaon, R. (2019). A novel approach for client side encryption in cloud computing. In 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE) (pp. 1–6). IEEE. https://www.researchgate.net/publication/332586477_A_Novel_Approach_for_Client_Side_Encryption_in_Cloud_Computing

[11] Panigrahi, R., Borah, S., Bhoi, A. K., & Mallick, P. K. (2020). Intrusion detection systems (IDS)—An overview with a generalized framework. In Advances in Intelligent Systems and Computing (pp. 145–158). Springer. [https://doi.org/10.1007/978-981-15-1451-7_11]

[12] Shallal, Q. M., Bokhari, M. U., & Tamandani, Y. (2016). Cloud computing service models: A comparative study. In Proceedings of the 2016 IEEE Conference on Cloud Computing. IEEE.

[13] Gao, J., Gan, L., Buschendorf, F., Zhang, L., Liu, H., Li, P., & Lu, T. (2021). Omni SCADA intrusion detection using deep learning algorithms. *IEEE Internet of Things Journal, 8*(2), 951–961. https://doi.org/10.1109/JIOT.2020.3009180

[14] Gao, Y., Liu, Y., Jin, Y., Chen, J., & Wu, H. (2018). A novel semi-supervised learning approach for network intrusion detection on cloud-based robotic system. *IEEE Access, 6,* 50927–50938. https://doi.org/10.1109/ACCESS.2018.2869457

[15] Gautam, R. K., & Doegar, E. A. (2018). An ensemble approach for intrusion detection system using machine learning algorithms. *Journal of Cloud Computing, Data Science and Engineering, 3,* 14–15.

[16] Geva, M., Herzberg, A., & Gev, Y. (2013). Bandwidth distributed denial of service: Attacks and defenses. *IEEE Security & Privacy, 12*(5), 54–61. https://doi.org/10.1109/MSP.2013.92

[17] Gnana Singh, D., & Leavline, E. J. (2013). Data mining in network security—Techniques and tools: A research perspective. *Journal of Theoretical and Applied Information Technology, 57,* 1–8.

[18] Jayapandian, S., Saidi, F., Trabelsi, Z., & Ben Ghazela, H. (2020). Fuzzy IDS as a service on the cloud for malicious TCP port scanning traffic detection. *International Journal on Digital Technologies and Applications (IJDTA).* https://doi.org/10.3233/IDT-180050

[19] Jonathan, R., Ramkumar, B. N., & Subbulakshmi, T. (2021). TCP SYN flood attack detection and prevention system using adaptive thresholding method. *ITM Web of Conferences, 37,* 01016. https://doi.org/10.1051/itmconf/20213701016

[20] Kulwinder, K., Saad, A., Mansour, M., & Atef, A. (2023). An adaptive distributed DDoS attack prevention technique in a distributed environment. *Sensors, 23*(14), 6574. https://www.mdpi.com/1424-8220/23/14/6574

[21] Maglaras, L., Ferrag, M. A., & Ahmed, J. (2020). AEGIS: Detection and mitigation of TCP SYN flood on SDN controller. *IEEE Transactions on Network and Service Management.* https://doi.org/10.1109/TNSM.2020.

[22] Mahrach, B., Bensaid, R., Labraoui, N., Ari, A. A. A., Maglaras, L., Saidi, H., & Lwahhab, A. M. (2023). Toward a real-time TCP SYN flood DDoS mitigation using adaptive neuro fuzzy classifier and SDN assistance in fog computing. *arXiv preprint arXiv:2311.15633.* https://arxiv.org/abs/2311.15633

[23] Mehmood, T., Bekravi, M., Jamali, S., & Shaker, G. (2012). Defense against SYN flood DoS attacks based on learning automata. *arXiv preprint arXiv:1208.5037.* https://arxiv.org/abs/1208.5037

[24] Oktay, A., Gupta, S., Kumar, P., & Abraham, A. (2013). Profile-based NIDS/NIPS for securing cloud environment. *International Journal of Distributed Sensor Networks, 2013,* Article 364575. https://doi.org/10.1155/2013/364575

[25] Said, H. M., Sharma, A., Islam, M. R., & Ningombam, D. (2021). Impact of TCP SYN flood attack in cloud. In *Contemporary issues in communication, cloud and big data analytics* (Lecture Notes in Networks and Systems, Vol. 281). Springer. https://doi.org/10.1007/978-981-16-4244-9_7

[26] Shaikh, M., Dang, V. T., et al. (2022). TFAD: TCP flooding attack detection in SDN using proxy-based and machine learning mechanisms. *Cluster Computing.* https://doi.org/10.1007/s10586-022-03666-4

[27] Valavan, W. T., Joseph, N., & Umarani, S. G. (2024). Intrusion detection system in cloud computing by utilizing VTR-HLSTM based on deep learning. *Indonesian Journal of Electrical Engineering and Computer Science, 33*(3), 1829–1838. https://www.researchgate.net/publication/378647717

[28] Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access, 5,* 21954–21961. https://doi.org/10.1109/ACCESS.2017.2762418