An Enhanced MCAD Framework using Ensemble Learning for Cyberattack Detection in Healthcare SDN

[1]SHAIK SHAGUFTA FATHIMA, [2]N MUNIKRISHNA

[1]PG Scholar, Dept. of CSE, KMM Institute of Technology & Science, Ramireddipalle, Tirupati, AP, India.

[2]Assistant Professor, Dept. of CSE, KMM Institute of Technology & Science, Ramireddipalle, Tirupati, AP, India.

Abstract: The healthcare sector relies on Software-Defined Networking (SDN) to manage sensitive patient data, but it remains highly vulnerable to diverse cyberattacks. To address this challenge, this paper presents an enhanced Machine Learning-based Cyberattack Detector (MCAD) that incorporates ensemble learning techniques, specifically Stacking and Voting classifiers, to improve detection accuracy and robustness. The system processes SDN traffic data through preprocessing, feature selection, and multiple machine learning models, whose outputs are combined to achieve optimal performance. Additionally, a secure Flask-based web application with user authentication is developed to enable real-time prediction and practical deployment in healthcare environments. Experimental results demonstrate that the proposed system achieves up to 100% accuracy with improved reliability, making it an effective solution for protecting healthcare SDN infrastructures against evolving cyber threats.

Index terms - *Cyberattack Detection, Software-Defined Networking (SDN), Healthcare Security, Machine Learning, Ensemble Learning, Stacking Classifier, Voting Classifier, Intrusion Detection System (IDS), Flask Web Application, Network Security.*

1. INTRODUCTION

Many industries have adopted SDNs in recent years. This is largely because they are reliable network technologies that separate the control and data planes for network management. SDN offers the controller and applications more network state knowledge than traditional networks, which just know about their apps. Due to the fast rise of information and communication technologies (ICT), healthcare organizations are using many of the same off-the-shelf technology, software, and procedures as other companies. We expected this since networked or Internet-connected medical systems improve asset management, communications, and electronic health records, saving money. In addition, most information systems prioritize system and device safety and user data privacy. Due to rigorous regulations, healthcare privacy and safety are crucial. Even though hospital equipment expenses are projected, the present McAfee record said that networked medical instruments may have security gaps as the medical industry seeks to integrate all the technical aspects of networked infrastructure and operational controls.

This project creates a software-defined network-based machine learning-based cyber-attack detector (MCAD) to protect healthcare systems. MCAD will look at routine and unusual network traffic using a layer three (L3) learning switch application on the Ryu controller. Testing with various machine learning algorithms and cyberattack scenarios offers a complete view of the system's performance. Its strong F1-score for normal and attack classes proves MCAD's reliability. Additionally, it can process 5,709,692 samples per second in real time.

The healthcare industry struggles to secure sensitive patient data in SDNs. SDNs offer many benefits, but they are vulnerable to several threats that might undermine the network and endanger patients. This work aims to construct a healthcare system machine learning-based cyber-attack detector (MCAD). This will be done using a Ryu controller layer three (L3) learning switch application. To increase healthcare data security and network resilience, this study will completely test MCAD's performance against a variety of machine learning algorithms and attack scenarios.

2. LITERATURE SURVEY

1. Intelligent Edge Load Migration in SDN-IIoT for Smart Healthcare:

Modern technology has caused greater healthcare issues. Sensors, IIoT, and big data analytics can enhance patient care and save healthcare costs. People will get safer, cheaper, and better medical care. IoT devices with insufficient resources, identity theft, and malicious insiders are problems. Edge computing services are needed to improve smart healthcare using big data and AI. These challenges may be addressed using an SDN-based security compliance framework for smart healthcare load transfer systems. Experts are studying ways to use SDN-IIoT to prevent security breaches in real time. We provide a three-domain framework. Domains have one virtual machine and numerous OpenFlow virtual switches. This shifts a lot of healthcare data from heavily loaded to lightly loaded. This balances the domain and prevents migration attacks. We use Wireshark to record OpenFlow packets from the RYU SDN controller to analyze the simulations and mininet performance. The framework allows secure data management, and the approach is 80% accurate for all healthcare data packets fetched.

2. Studying the effect of internal DOS attacks over SDN controller during switch registration process:

Numerous benefits are offered by software-defined networks, including centralization, application programmability interfaces, and the capacity to swiftly implement policies throughout the whole network. Networks with centralized control are more secure and scalable than traditional networks. It may, however, also be more susceptible to denial-of-service attacks from both inside and outside the network. This article compares two of the most popular SDN controllers and shows how an internal denial of service attack might affect the southbound interface during switch registration. The controller's CPU use and reaction time are tracked and examined during the attack.

3. Intruder Detection System Based Artificial Neural Network for Software Defined Network:

This article explains how to set up an Intruder Detection System (IDS) that uses an Artificial Neural Network (ANN) called Snort + RNA to reduce the dangers of active computer attacks on a Software Defined Network (SDN). It utilizes the hyperconverged network of the Technical University of the North's Faculty of Engineering of Applied Science (FICA). This idea is tested using the hacking circle's techniques and the PDCA model from the ISO/IEC 27001 standard. The findings demonstrate that Snort + RNA may identify oddities that result in SDN active-type assaults. Both the record of the captured traffic and the alerts that are sent out make this evident. However, because some of the packets it receives from DoS assaults are still on hold or have been denied, it is not able to examine them all. This demonstrates how the system protects the SDN by providing alerts when third parties attempt to attack it, which increases network traffic, even if it doesn't examine every packet that travels across the network.

4. Survey on Intrusion Detection System in IoT Network:

The Internet of Things (IoT) has developed into a powerful means of communication and connection between automated processes and smart gadgets. As the Internet of Things is used for more and more crucial functions, it's critical to ensure that the devices can communicate with one another in a safe and secure manner. The biggest threat to secure and safe communication is cyberattacks. Cyberattacks have gotten more sophisticated and harder to stop, making it harder to keep data safe, communicate securely, and keep data private. Because they can identify security flaws in a communication network, intrusion detection systems are the ideal method for safeguarding IoT devices. However, integrating an intrusion detection system into an Internet of Things network is challenging. By examining significant previous research, this paper examines the most cutting-edge IoT and intrusion detection system technologies, the technologies now in use, and the issues that arise with these technologies. We examined 25 sources, including 22 scholarly publications and papers, that discussed threat models, important challenges for Internet of Things (IoT) intrusion detection systems, suggested models, and how to implement such models. The findings examine the needs and optimal methods for integrating AI-based intrusion detection systems into IoT networks to maintain secure communication.

5. Intrusion Detection Systems in Internet of Things and Mobile Ad-Hoc Networks:

Since the majority of Internet of Things (IoT) devices operate wirelessly, they require various types of Intrusion Detection Systems (IDS) to identify intrusions using 802.11 header information. at wired networks, the majority of wireless-specific traffic characteristics with significant information gain may be found at data link levels rather than application layers. This article examines some of the challenges and issues associated with wireless intrusion detection systems (IDS), including data collection, IDS technology application, IDS placement, and traffic data analysis. The main finding of this study is that there aren't enough network traces available to train modern machine-learning algorithms to handle intrusions unique to the Internet of Things. The Knowledge Discovery in Databases (KDD) Cup dataset is examined in order to illustrate the design issues of wireless intrusion detection based on current data attributes. Additionally, the study proposes certain guidelines to ensure that future wireless network (WN) traffic capture techniques continue to function. The paper starts by discussing several methods for detecting intrusions, collecting information, and placing it appropriately. Examining the issues that arise while creating an intrusion detection system for a wireless environment is the main goal of this study. Because of the architectural challenges, setting up an intrusion detection system in a wireless environment is more difficult than in a wired network. This article examines how cable intrusion detection systems are typically configured and discusses how comparable techniques may be applied in a wireless environment. The design issues that arise in a wireless environment are also discussed. The Internet of Things (IoT), Mobile Ad Hoc Networks (MANET), and Wireless Sensor Networks (WSN) are the most important wireless environments to examine as they are the future and have been the target of numerous attacks. Making an IDS that only works on wireless networks is crucial.

3. METHODOLOGY

i) Proposed Work:

In order to improve cybersecurity in healthcare systems that depend on Software-Defined Networking (SDN), the suggested solution incorporates MCAD (Machine Learning-based Cyberattack Detector). By implementing an intelligent detection mechanism directly on the Ryu controller, MCAD takes advantage of the centralized control of SDN. It use machine learning algorithms to categorize and react to various cyberthreats in real-time after capturing both legitimate and malicious data via a Layer 3 learning switch. MCAD guarantees prompt threat detection and aids in preserving safe access to sensitive patient data, which is essential in healthcare settings, by examining a variety of attack patterns and traffic trends.

In order to provide high detection accuracy, flexibility, and quick performance, MCAD is trained on a variety of attack datasets and supports a broad range of machine learning models. Because of its architecture, which enables real-time threat mitigation, the system is appropriate for settings where data integrity and uptime are critical. MCAD exhibits efficiency and dependability with its capacity to handle more than 5 million samples per second and provide a high F1-score in classification. By overcoming the drawbacks of conventional static security solutions like signature-based IDS, this clever, data-driven method enables healthcare SDN systems to identify and react to changing cyber threats dynamically.

ii) System Architecture:

The first step in the MCAD system design is data collection from the network, where topology-based traffic is gathered. To get ready for machine learning, this raw data goes through a number of data preparation procedures, such as cleansing, feature modification, scaling, and shuffling. Following preprocessing, the data is divided into three sets: testing, validation, and training. To create an optimal machine learning model, the training procedure entails model training and hyperparameter tweaking. The Ryu SDN controller then uses this trained model to continually classify incoming unknown network traffic as either regular or attack in real-time. For SDN-based healthcare systems, this architecture guarantees precise, flexible, and effective cyberattack detection.

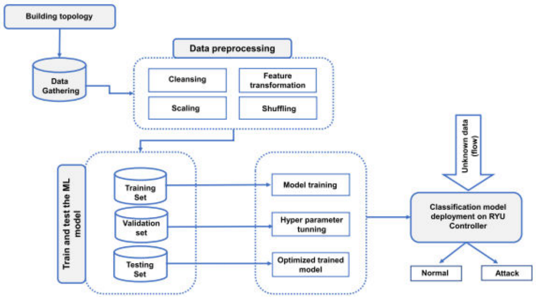


Fig.1. Proposed Architecture

iii) MODULES:

a. Proposing a Logical Network Topology: The first step in the model is to come up with a logical network topology for the healthcare system.

b. Data Gathering: The model gathers information to train and evaluate the machine learning (ML) model [19,42]. This includes regular samples as well as several forms of attacks, such as probing assaults, exploiting the VNC port 5900 remote view vulnerability, and exploiting the Samba server vulnerability.

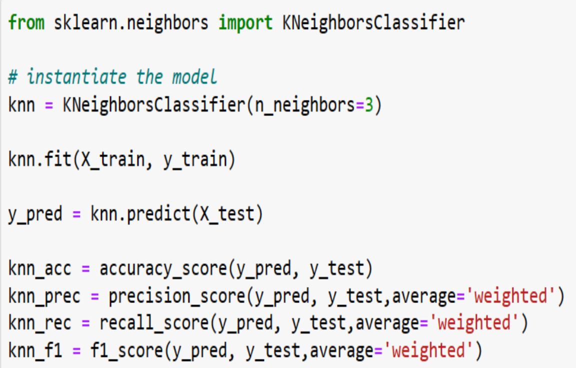
c. Data Preprocessing: The data that was gathered is cleaned up so that it may be used to train the ML model.

d. Training and Testing the ML Model: We use different classification techniques to train and evaluate the ML model. These include KNN, decision tree (DT), random forest (RF), naïve Bayes (NB), logistic regression (LR), adaptive boosting (adaboost), and xgboost (XGB). The model produces a mapping function between inputs and outputs by finding patterns and reducing mistakes. Accuracy is used to measure performance [19,42].

e. Deployment of the project : The trained ML model is put to use on the user interface. This makes it possible to use the model in real-time systems, which helps keep the healthcare system's quality high.

iv) ALGORITHMS:

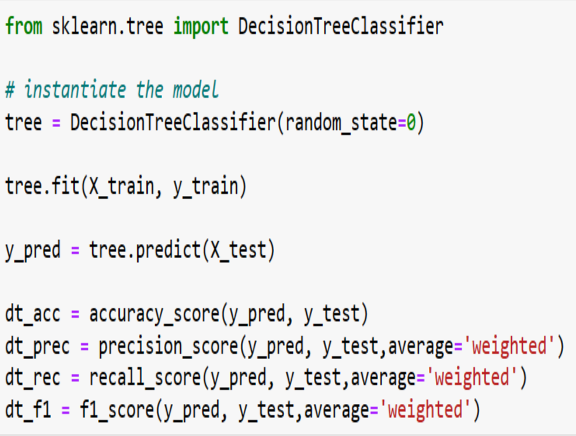
K Nearest Neighbour: KNN is a supervised method that may be used for both classification and regression. It sorts data into groups based on the majority class of their k-nearest neighbours (k is set by the user), assuming that comparable data points are nearby to each other in the feature space. You can use KNN to sort out different types of network traffic in a healthcare SDN context [1,8,12]. By comparing patterns to known cases, it helps find unusual behaviour.



```
from sklearn.neighbors import KNeighborsClassifier

# instantiate the model
knn = KNeighborsClassifier(n_neighbors=3)

knn.fit(X_train, y_train)

y_pred = knn.predict(X_test)

knn_acc = accuracy_score(y_pred, y_test)
knn_prec = precision_score(y_pred, y_test,average='weighted')
knn_rec = recall_score(y_pred, y_test,average='weighted')
knn_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 3 KNN

Decision trees: you may use decision trees for both classification and regression. They're like trees, with nodes that test features and branches that lead to results. They make choices by moving from the root to the leaves based on input features. You can use decision trees to construct rules for finding strange things on a network. Decision trees are useful for figuring out how the network works since they are easy to grasp.

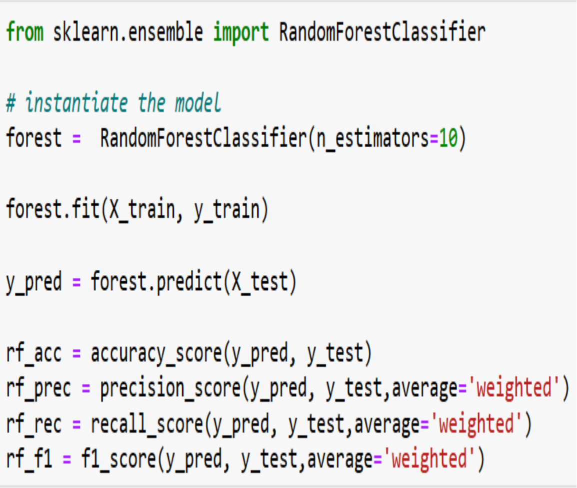


```
from sklearn.tree import DecisionTreeClassifier

# instantiate the model
tree = DecisionTreeClassifier(random_state=0)

tree.fit(X_train, y_train)

y_pred = tree.predict(X_test)

dt_acc = accuracy_score(y_pred, y_test)
dt_prec = precision_score(y_pred, y_test,average='weighted')
dt_rec = recall_score(y_pred, y_test,average='weighted')
dt_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 4 Decision tree

Random Forest: Random Forest is a system that combines several decision trees into a single forest.

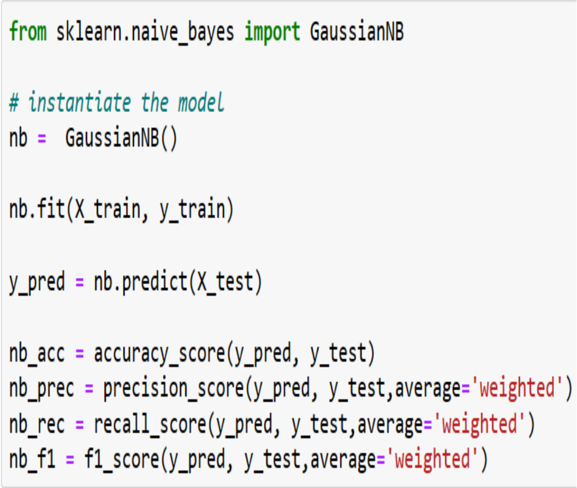
You may make forecasts by averaging or voting on the trees' projections. It helps reduce overfitting and makes the model more accurate. By combining predictions from numerous decision trees, Random Forest can make cyberattack detection more reliable. It helps reduce the number of false positives and false negatives in healthcare network security [24], [28], and [30].



```
from sklearn.ensemble import RandomForestClassifier

# instantiate the model
forest =  RandomForestClassifier(n_estimators=10)

forest.fit(X_train, y_train)

y_pred = forest.predict(X_test)

rf_acc = accuracy_score(y_pred, y_test)
rf_prec = precision_score(y_pred, y_test,average='weighted')
rf_rec = recall_score(y_pred, y_test,average='weighted')
rf_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 5 Random forest

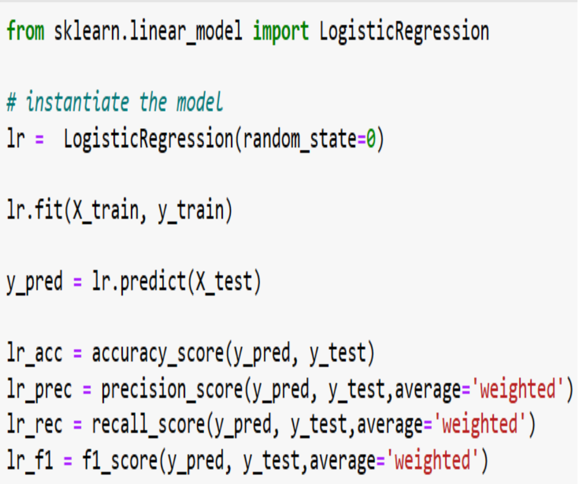
Naive Bayes: Naive Bayes is a type of probabilistic classifier that uses Bayes' theorem. It makes things easier by assuming that characteristics are conditionally independent, which is a common technique for classifying text and filtering spam. Naive Bayes can help with text categorisation, which is vital for finding bad traffic in healthcare communication. It may be used to find strange patterns in text in network data [54].



```
from sklearn.naive_bayes import GaussianNB

# instantiate the model
nb =  GaussianNB()

nb.fit(X_train, y_train)

y_pred = nb.predict(X_test)

nb_acc = accuracy_score(y_pred, y_test)
nb_prec = precision_score(y_pred, y_test,average='weighted')
nb_rec = recall_score(y_pred, y_test,average='weighted')
nb_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 6 Naïve bayes

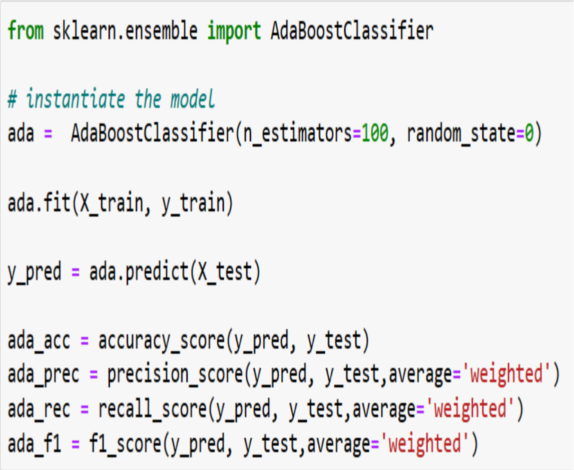
Logistic Regression: Logistic Regression is a type of statistical model that is used to solve issues with two possible outcomes. It gives an estimate of how likely it is that a certain input belongs to a certain class. The logistic function is used to model the connection between the dependent variable (binary result) and one or more independent factors. Logistic Regression may help figure out how likely it is that network events are connected to cyberattacks, which makes it useful for binary categorisation in healthcare network security [55].



```
from sklearn.linear_model import LogisticRegression

# instantiate the model
lr =  LogisticRegression(random_state=0)

lr.fit(X_train, y_train)

y_pred = lr.predict(X_test)

lr_acc = accuracy_score(y_pred, y_test)
lr_prec = precision_score(y_pred, y_test,average='weighted')
lr_rec = recall_score(y_pred, y_test,average='weighted')
lr_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 7 Logistic regression

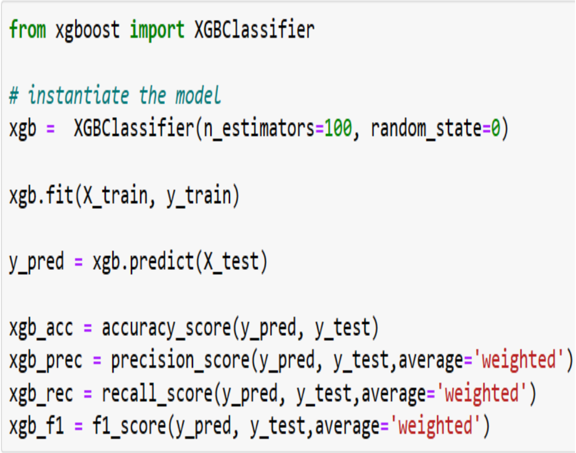
Adaboost: Adaboost is a way to combine weak classifiers to make a powerful one. It focusses on instances that are incorrectly categorised, which lets later classifiers fix mistakes. A lot of the time, it's utilised for binary categorisation. Adaboost can make basic classifiers work better, which makes it a great tool for making cyberattack detection in healthcare SDNs more accurate [56].



```
from sklearn.ensemble import AdaBoostClassifier

# instantiate the model
ada =  AdaBoostClassifier(n_estimators=100, random_state=0)

ada.fit(X_train, y_train)

y_pred = ada.predict(X_test)

ada_acc = accuracy_score(y_pred, y_test)
ada_prec = precision_score(y_pred, y_test,average='weighted')
ada_rec = recall_score(y_pred, y_test,average='weighted')
ada_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 8 Adaboost

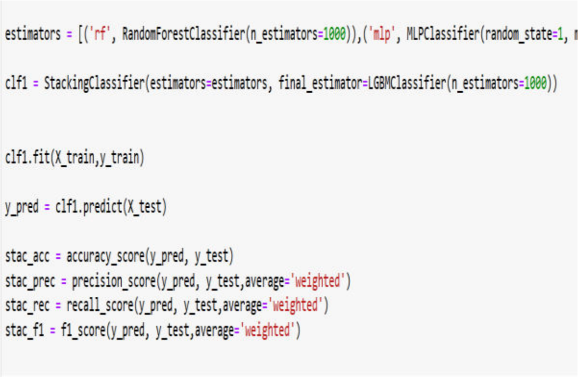
XGBoost: XGBoost is a supervised learning method that uses gradient boosting and is recognised for being fast, accurate, able to handle missing data, and able to process data in parallel. In machine learning contests and applications, it's quite popular. You may use XGBoost, which is recognised for its excellent accuracy, to construct a strong and dependable model for detecting cyberattacks. This will keep healthcare data as safe as possible.



```
from xgboost import XGBClassifier

# instantiate the model
xgb =  XGBClassifier(n_estimators=100, random_state=0)

xgb.fit(X_train, y_train)

y_pred = xgb.predict(X_test)

xgb_acc = accuracy_score(y_pred, y_test)
xgb_prec = precision_score(y_pred, y_test,average='weighted')
xgb_rec = recall_score(y_pred, y_test,average='weighted')
xgb_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 9 XGBoost

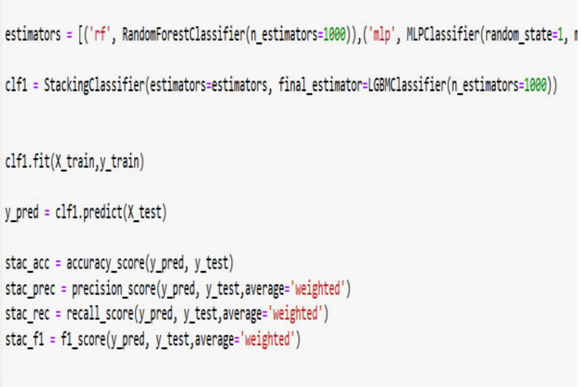
Stacking: Stacking employs a meta-learner that takes the outputs of base classifiers and makes final predictions. This improves the predictive performance of basic classifiers. It improves accuracy by finding a wider range of patterns. Stacking may be used to construct a group of different cyberattack detection models that can find a wide range of attack patterns and make healthcare systems safer overall.



```
estimators = [('rf', RandomForestClassifier(n_estimators=1000)),('mlp', MLPClassifier(random_state=1,

clf1 = StackingClassifier(estimators=estimators, final_estimator=LGBMClassifier(n_estimators=1000))


clf1.fit(X_train,y_train)

y_pred = clf1.predict(X_test)

stac_acc = accuracy_score(y_pred, y_test)
stac_prec = precision_score(y_pred, y_test,average='weighted')
stac_rec = recall_score(y_pred, y_test,average='weighted')
stac_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 10 Stacking classifier

Voting: Voting is a process that combines the predictions of many basic classifiers. It might be hard (majority vote) or soft (class probabilities). Voting classifiers make models more resilient and accurate by using the best parts of several models. You may use a voting classifier to integrate the judgements of many detection models. This makes it easier to find cyberattacks in the healthcare network that are more reliable and strong..



```
estimators = [('rf', RandomForestClassifier(n_estimators=1000)),('mlp', MLPClassifier(random_state=1,

clf1 = StackingClassifier(estimators=estimators, final_estimator=LGBMClassifier(n_estimators=1000))


clf1.fit(X_train,y_train)

y_pred = clf1.predict(X_test)

stac_acc = accuracy_score(y_pred, y_test)
stac_prec = precision_score(y_pred, y_test,average='weighted')
stac_rec = recall_score(y_pred, y_test,average='weighted')
stac_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 11 Voting classifier

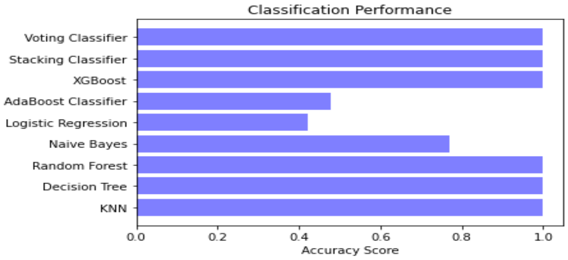
4. EXPERIMENTAL RESULTS

To ensure robustness and flexibility, a variety of machine learning methods and cyberattack scenarios were used in the experimental evaluation of the proposed MCAD system. Each model was thoroughly evaluated to determine its efficacy in identifying both typical and malicious traffic after the dataset was split into training, validation, and testing sets. MCAD outperformed the other models in terms of classification ability and reliability, as evidenced by its high F1-score for both attack and normal classes. Additionally, the system's remarkable throughput performance of 5,709,692 samples per second revealed its capacity to function effectively in real-time healthcare settings. These findings demonstrate that MCAD not only offers precise identification but also guarantees low latency and fast processing, both of which are essential for preserving the security and functionality of healthcare networks.

Accuracy: The ability of a test to differentiate between healthy and sick instances is a measure of its accuracy. Find the proportion of analysed cases with true positives and true negatives to get a sense of the test's accuracy. Based on the calculations:

Accuracy = TP + TN /(TP + TN + FP + FN)

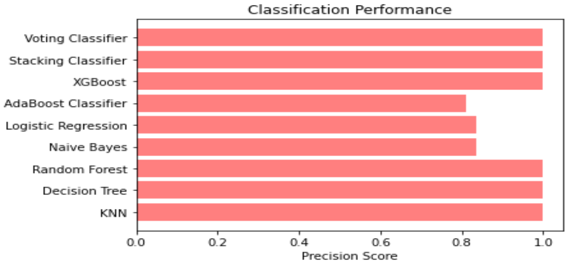
$$Accuracy = \frac{(TN + TP)}{T}$$



Precision: The accuracy rate of a classification or number of positive cases is known as precision. Accuracy is determined by applying the following formula:

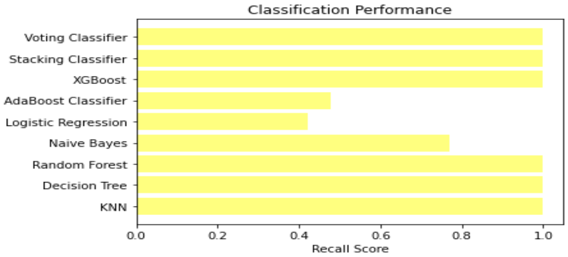
Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\mathrm{Pr}\,e\,cision = \frac{TP}{(TP+FP)}$$



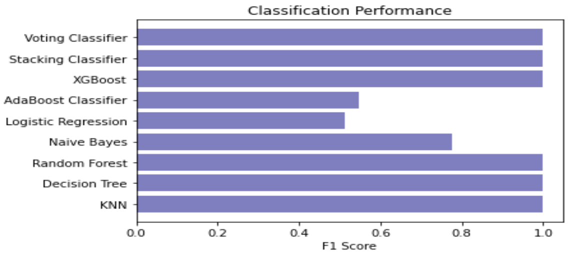
Recall: The recall of a model is a measure of its capacity to identify all occurrences of a relevant machine learning class. A model's ability to detect class instances is shown by the ratio of correctly predicted positive observations to the total number of positives.

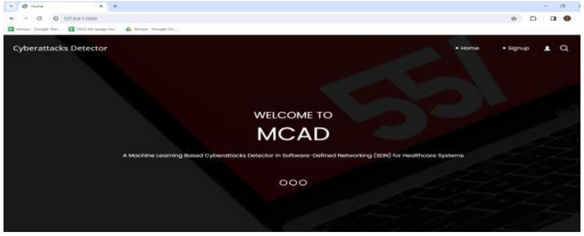
$$Recall = \frac{TP}{(FN+TP)}$$



F1-Score: A high F1 score indicates that a machine learning model is accurate. Improving model accuracy by integrating recall and precision. How often a model gets a dataset prediction right is measured by the accuracy statistic..

$$F1 = 2 \cdot \frac{(Recall \cdot \mathrm{Pr}\,e\,cision)}{(Recall + \mathrm{Pr}\,e\,cision)}$$



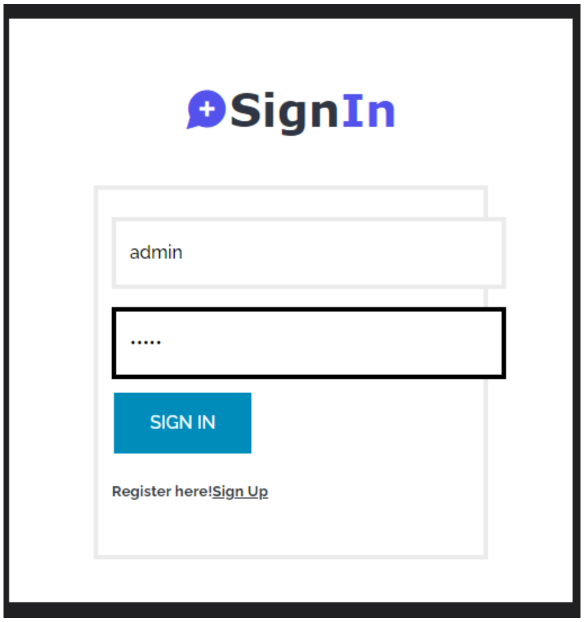


WELCOME TO

MCAD

A Machine Learning Based Cyberattacks Detector in Software-Defined Networking (SDN) for Healthcare Systems

Fig2 home screen

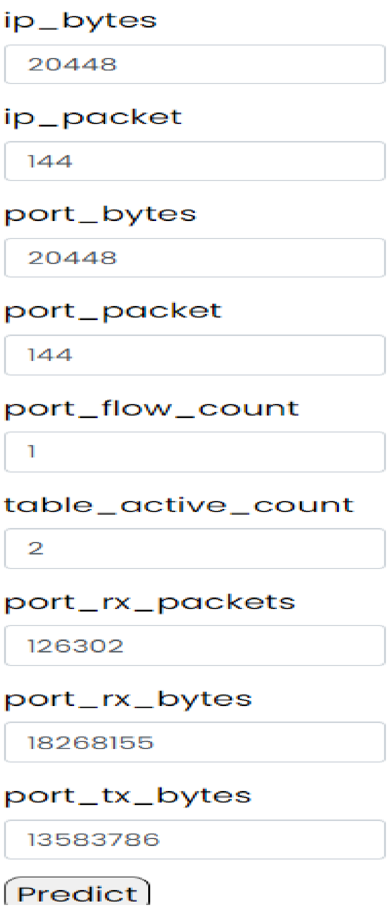


SignIn

admin

•••••

Fig 3 User Login



ip_bytes: 20448

ip_packet: 144

port_bytes: 20448

port_packet: 144

port_flow_count: 1

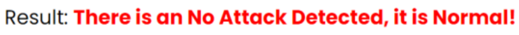
table_active_count: 2

port_rx_packets: 126302

port_rx_bytes: 18268155

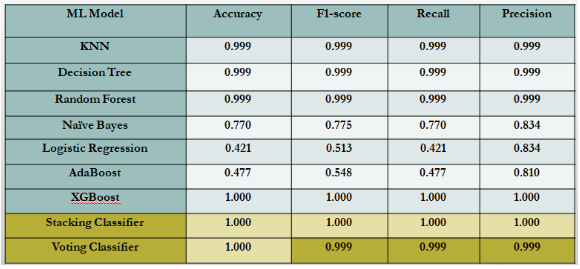
port_tx_bytes: 13583786

Fig4 User input page



Result: There is an No Attack Detected, it is Normal!

Fig5 Prediction result



ML Model	Accuracy	F1-score	Recall	Precision
KNN	0.999	0.999	0.999	0.999
Decision Tree	0.999	0.999	0.999	0.999
Random Forest	0.999	0.999	0.999	0.999
Naïve Bayes	0.770	0.775	0.770	0.834
Logistic Regression	0.421	0.513	0.421	0.834
AdaBoost	0.477	0.548	0.477	0.810
XGBoost	1.000	1.000	1.000	1.000
Stacking Classifier	1.000	1.000	1.000	1.000
Voting Classifier	1.000	0.999	0.999	0.999

Fig6 Performance Comparison of Forecasting Algorithms

5. CONCLUSION

By incorporating machine learning-based threat detection, the suggested MCAD system successfully improves the cybersecurity of healthcare systems using Software-Defined Networking. MCAD quickly and precisely detects both known and unexpected cyberattacks by utilizing real-time traffic analysis and implementing the optimal model on the Ryu controller. High F1-scores and outstanding throughput attest to the system's robust performance, which validates its dependability and effectiveness in meeting the demanding requirements of healthcare settings. All things considered, MCAD provides a high-performance, scalable, and adaptable solution for safeguarding private medical information and guaranteeing continuous healthcare services.

6. FUTURE SCOPE

Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are two deep learning approaches that can be integrated into the MCAD system in the future to increase detection accuracy for complex and dynamic cyber threats. Furthermore, the detection system may be updated in real time without having to retrain from scratch by utilizing adaptive models and online learning. Additionally, the system may be extended to provide cross-domain threat detection, which would make it useful in industries other than healthcare, including banking or industrial IoT networks. Additionally, including automated response systems and blockchain-based tracking for audit trails can greatly improve network resilience and event handling.

REFERENCES

[1] M. Jarschel, T. Zinner, T. Hossfeld, P. Tran-Gia, and W. Kellerer, ''Interfaces, attributes, and use cases: A compass for SDN,'' IEEE Commun. Mag., vol. 52, no. 6, pp. 210–217, Jun. 2014.

[2] W. Meng, K.-K.-R. Choo, S. Furnell, A. V. Vasilakos, and C. W. Probst, ''Towards Bayesian-based trust management for insider attacks in healthcare software-defined networks,'' IEEE Trans. Netw. Service Manage., vol. 15, no. 2, pp. 761–773, Jun. 2018.

[3] J. T. Kelly, K. L. Campbell, E. Gong, and P. Scuffham, ''The Internet of Things: Impact and implications for health care delivery,'' J. Med. Internet Res., vol. 22, p. 11, Nov. 2020.

[4] (2022). Networked Medical Devices: Security and Privacy Threats—Sym antec—[PDF Document]. [Online]. Available: https://fdocuments. net/document/networked-medical-devices-security-and-privacy-threatssymantec.html

[5] P. A. Williams and A. J. Woodward, ''Cybersecurity vulnerabilities in medical devices: A complex environment and multifaceted problem,'' Med. Devices, Evidence Res., vol. 8, pp. 305–316, Jul. 2015.

[6] C. M. Williams, R. Chaturvedi, and K. Chakravarthy, ''Cybersecurity risks in a pandemic,'' J. Med. Internet Res., vol. 22, no. 9, Sep. 2020, Art. no. e23692.

[7] N. Thamer and R. Alubady, ''A survey of ransomware attacks for healthcare systems: Risks, challenges, solutions and opportunity of research,'' in Proc. 1st Babylon Int. Conf. Inf. Technol. Sci. (BICITS), I. Babil, Ed., Apr. 2021, pp. 210–216.

[8] H. Babbar, S. Rani, and S. A. AlQahtani, ''Intelligent edge load migration in SDN-IIoT for smart healthcare,'' IEEE Trans. Ind. Informat., vol. 18, no. 11, pp. 8058–8064, Nov. 2022.

[9] R. Hasan, S. Zawoad, S. Noor, M. M. Haque, and D. Burke, ''How secure is the healthcare network from insider attacks? An audit guideline for vulnerability analysis,'' in Proc. IEEE 40th Annu. Comput. Softw. Appl. Conf. (COMPSAC), Jun. 2016, pp. 417–422.

[10] (Apr. 2015). 92% of Healthcare IT Admins Fear Insider Threats Thales. Accessed: Mar. 21, 2023. [Online]. Available: https://cpl.thalesgroup. com/about-us/newsroom/news-releases/92-healthcare-it-admins-fearinsider-threats

[11] D. Chaulagain, K. Pudashine, R. Paudyal, S. Mishra, and S. Shakya, ''OpenFlow-based dynamic traffic distribution in software-defined networks,'' in Mobile Computing and Sustainable Informatics. Singapore: Springer, Jul. 2021, pp. 259–272.

[12] R. Khondoker, A. Zaalouk, R. Marx, and K. Bayarou, ''Feature-based comparison and selection of software defined networking (SDN) controllers,'' in Proc. World Congr. Comput. Appl. Inf. Syst. (WCCAIS), Jan. 2014, pp. 1–7.

[13] T. Mekki, I. Jabri, A. Rachedi, and L. Chaari, ''Software-defined networking in vehicular networks: A survey,'' Trans. Emerg. Telecommun. Technol., vol. 33, no. 10, pp. 1–10, Apr. 2021, doi: 10.1002/ett.4265.

[14] Z. Ghaffar, A. Alshahrani, M. Fayaz, A. M. Alghamdi, and J. Gwak, ''A topical review on machine learning, software defined networking, Internet of Things applications: Research limitations and challenges,'' Electronics, vol. 10, no. 8, p. 880, Apr. 2021, doi: 10.3390/electronics10080880.

[15] C.-S. Li and W. Liao, ''Software defined networks [guest editorial],'' IEEE Commun. Mag., vol. 51, no. 2, p. 113, Feb. 2013.

[16] M. H. Rehmani, A. Davy, B. Jennings, and C. Assi, ''Software defined networks-based smart grid communication: A comprehensive survey,'' IEEE Commun. Surveys Tuts., vol. 21, no. 3, pp. 2637–2670, 3rd Quart., 2019.

[17] L. F. Eliyan and R. Di Pietro, ''DoS and DDoS attacks in software defined networks: A survey of

existing solutions and research challenges,'' Future Gener. Comput. Syst., vol. 122, pp. 149–171, Sep. 2021, doi: 10.1016/j.future.2021.03.011.

[18] K. Benton, L. J. Camp, and C. Small, ``OpenFlow vulnerability assessment,'' in Proc. 2nd ACM SIGCOMM Workshop Hot Topics Softw. Defined Netw., 2013, pp. 151–152, doi: 10.1145/2491185.2491222.

[19] B. Mladenov and G. Iliev, ``Studying the effect of internal DOS attacks over SDN controller during switch registration process,'' in Proc. Int. Symp. Netw., Comput. Commun. (ISNCC), Jul. 2022, pp. 1–4.

[20] H. Domínguez-Limaico, W. N. Quilca, M. Zambrano, F. Cuzme-Rodríguez, and E. Maya-Olalla, ``Intruder detection system based artificial neural network for software defined network,'' in Proc. Int. Conf. Technol. Res. Cham, Switzerland: Springer, Aug. 2022, pp. 315–328.

[21] S. A. Mehdi and S. Z. Hussain, ``Survey on intrusion detection system in IoT network,'' in Proc. Int. Conf. Innov. Comput. Commun. Singapore: Springer, Sep. 2022, pp. 721–732.

[22] V. Ponnusamy, M. Humayun, N. Z. Jhanjhi, A. Yichiet, and M. F. Almufareh, ``Intrusion detection systems in Internet of Things and mobile ad-hoc networks,'' Comput. Syst. Sci. Eng., vol. 40, no. 3, pp. 1199–1215, 2022, doi: 10.32604/csse.2022.018518.

[23] K. Malasri and L. Wang, ``Securing wireless implantable devices for healthcare: Ideas and challenges,'' IEEE Commun. Mag., vol. 47, no. 7, pp. 74–80, Jul. 2009.

[24] D. Yin, L. Zhang, and K. Yang, ``A DDoS attack detection and mitigation with software-defined Internet of Things framework,'' IEEE Access, vol. 6, pp. 24694–24705, 2018.

[25] R. Wang, Z. Jia, and L. Ju, ``An entropy-based distributed DDoS detection mechanism in software-defined networking,'' in Proc. IEEE Trustcom/BigDataSE/ISPA, vol. 1, Aug. 2015, pp. 310–317.

[26] S. M. Mousavi and M. St-Hilaire, ``Early detection of DDoS attacks against SDN controllers,'' in Proc. Int. Conf. Comput., Netw. Commun. (ICNC), Feb. 2015, pp. 77–81.

[27] S. Murtuza and K. Asawa, ``Mitigation and detection of DDoS attacks in software defined networks,'' in Proc. 11th Int. Conf. Contemp. Comput., Aug. 2018, pp. 1–3.

[28] X. You, Y. Feng, and K. Sakurai, ``Packet in message based DDoS attack detection in SDN network using OpenFlow,'' in Proc. 5th Int. Symp. Comput. Netw. (CANDAR), Nov. 2017, pp. 522–528.

[29] S. Y. Mehr and B. Ramamurthy, ``An SVM based DDoS attack detection method for Ryu SDN controller,'' in Proc. 15th Int. Conf. Emerg. Netw. Exp. Technol., New York, NY, USA, Dec. 2019, pp. 72–73, doi: 10.1145/3360468.3368183.

[30] Q. Niyaz, W. Sun, and A. Y. Javaid, ``A deep learning based DDoS detection system in software-defined networking (SDN),'' ICST Trans. Secur. Saf., vol. 4, no. 12, Dec. 2017, Art. no. 153515. [Online]. Available: https://publications.eai.eu/index.php/sesa/article/view/211