IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.53

Volume 4, Issue 1, April 2024

Anomaly Detection System

Prof. K. G. Jagtap[1], Shreeram Shinde[2], Akshay Adhav[3], Sahil Kadambande[4]

Professor, Department of AI & ML [1]
Student, Department of AI & ML[2,3,4]
AISSMS Polytechnic, Pune, India

Abstract*: An "Anomaly Detection System" is an software/hardware application that monitors the activity of an network and alerts the user, of any activity that seems to be malicious or strange. The classification between an malicious traffic and normal traffic is done based on the set of rules that are pre-defined. An Anomaly Detection often referred to an IDS reduces the work performed by analyst to monitor network and automates the process of monitoring network traffic to detect anomaly in network traffic data. The implemented Anomaly Detection System takes into consideration various machine learning algorithms and detects abnormal traffic which can also be called as anomaly.*

Keywords: Anomaly, Anomaly Detection System, Denial Of Service, Random Forest, KNN Algorithm, Support Vector Machine.

I. INTRODUCTION

The implementation of an Anomaly Detection System (ADS) using machine learning techniques represents a proactive approach towards safeguarding network security in the face of escalating threats posed by malicious actors. In this endeavor, a multifaceted strategy is adopted, wherein diverse machine learning algorithms are harnessed to train the model, enabling it to effectively discern aberrant user activities within the network.Central to the development of this ADS is the utilization of a comprehensive dataset of network traffic, which serves as the bedrock for training the model. To ensure the efficacy of the model, several critical stages are meticulously undertaken. Initially, the raw data undergoes preprocessing, a crucial step aimed at cleansing and structuring the dataset to render it amenable for subsequent analysis. This entails tasks such as data cleaning, normalization, and possibly dimensionality reduction, all geared towards enhancing the quality and usability of the dataset. Subsequently, feature engineering assumes paramount importance, as it entails the extraction and selection of pertinent features from the dataset. This process involves identifying and isolating key attributes that encapsulate meaningful information regarding network behavior. These features serve as the foundation upon which the model will be trained, hence necessitating careful consideration and deliberation.

The crux of the ADS implementation lies in the judicious selection of machine learning algorithms, which are instrumental in discerning anomalous activities amidst the network traffic. By leveraging a diverse array of algorithms, ranging from traditional statistical methods to cutting-edge deep learning techniques, the ADS endeavors to comprehensively capture and mitigate potential threats. The rationale behind this approach lies in the recognition that different algorithms exhibit varying degrees of efficacy in detecting anomalies within distinct contexts. Therefore, by exploring a multitude of algorithms, the ADS endeavors to identify the most suitable algorithm for the specific characteristics of the dataset, thereby optimizing detection accuracy.The culmination of this endeavor involves rigorous evaluation of the trained model to ascertain its effectiveness in detecting anomalous network behavior. This entails subjecting the model to a battery of performance metrics, such as precision, recall, and F1-score, to gauge its efficacy under real-world conditions. Additionally, the model undergoes rigorous testing against a separate validation dataset to assess its generalization capability and resilience to unseen threats.

In essence, the implementation of an ADS using machine learning techniques represents a proactive and adaptive approach towards bolstering network security. By harnessing the power of data-driven insights and sophisticated algorithms, the proposed system.

DOI: 10.48175/568

II. EXISTING SYSTEM

In today's digital landscape, safeguarding the integrity and security of IT infrastructure stands as a paramount concern for organizations across industries. An Anomaly Detection System (ADS) emerges as a cornerstone in this endeavor, serving as a crucial line of defense against malicious cyber threats. While the performance capabilities of ADS may vary depending on the specific business rules and configurations implemented, its overarching goal remains consistent: to detect and mitigate potential security breaches and unauthorized access attempts. While acknowledging that no system can guarantee absolute protection against all conceivable threats, an ADS excels in its ability to identify and thwart a significant portion of malicious activities. By leveraging sophisticated algorithms and machine learning techniques, ADS can effectively analyze network traffic patterns and user behaviors to pinpoint anomalies indicative of potential security breaches. This proactive approach empowers organizations to preemptively address security vulnerabilities and fortify their IT infrastructure against a wide array of cyber threats.

Crucially, the significance of ADS extends beyond mere protection of organizational assets; it also plays a pivotal role in safeguarding sensitive user information. From personal details such as addresses and phone numbers to highly confidential financial data, ADS functions as a vigilant guardian, ensuring the confidentiality and privacy of user data remains uncompromised. By detecting and thwarting unauthorized access attempts, ADS mitigates the risk of data breaches and unauthorized disclosures, thereby fostering trust and confidence among users.

It's worth noting that the deployment of ADS can manifest in various forms, ranging from software-based solutions integrated within existing IT systems to dedicated hardware appliances tailored to specific organizational needs. Regardless of the implementation format, the core objective remains steadfast: to fortify the IT infrastructure and protect user data from malicious threats.

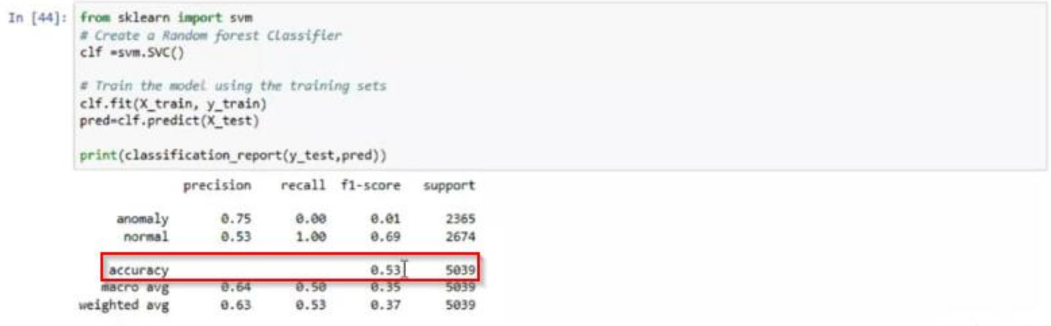
In essence, an Anomaly Detection System embodies the proactive stance adopted by organizations in safeguarding their digital assets and user information in an increasingly interconnected and digitized world. By leveraging advanced technology and robust security protocols, ADS stands as a bulwark against cyber threats, bolstering the resilience and security posture of organizations across the globe.

III. IMPLEMENTATION

The “Anomaly Detection System” implemented in this case is based on multiple machine learning algorithms, that help classify the network data into normal traffic and abnormal traffic also referred to as anomaly. The system takes into consideration three different algorithms which are: 1) SVM (Support Vector Machine), 2) KNN (K-Nearest Neighbor) and 3) Random Forest. The system tests the result against these algorithms and takes into consideration the algorithm that gives the best, accurate and fast results.

Initially the data is divided into two parts which are for training and testing into 80% and 20% respectively. Once the model is completely trained the system thoroughly test’s the model against labeled inputs to verify the performance of the system. Based on the dataset that we take into consideration that is KDDCUP1999 below are the measured results against the three different algorithms:

SVM Algorithm: The SVM algorithm gives an accuracy of 53 % when tested against the provided dataset. This is not the ideal result, thus some other techniques are also considered.



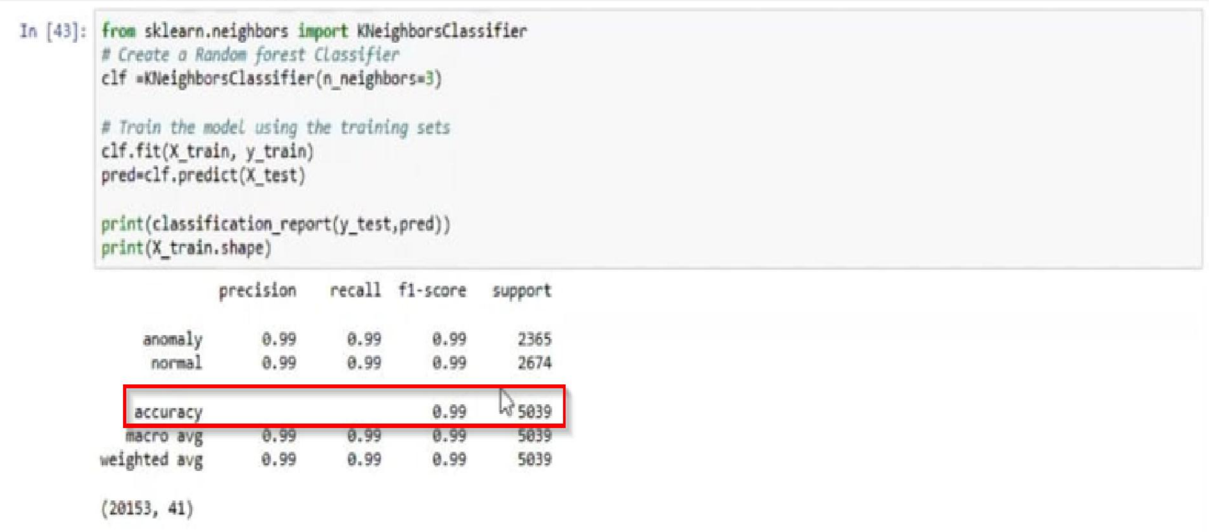
```
In [44]: from sklearn import svm
         # Create a Random forest Classifier
         clf =svm.SVC()

         # Train the model using the training sets
         clf.fit(X_train, y_train)
         pred=clf.predict(X_test)

         print(classification_report(y_test,pred))
```

```
              precision    recall  f1-score   support

     anomaly       0.75      0.00      0.01      2365
      normal       0.53      1.00      0.69      2674

    accuracy                           0.53      5039
   macro avg       0.64      0.50      0.35      5039
weighted avg       0.63      0.53      0.37      5039
```

Figure 1: Accuracy of SVM algorithm

DOI: 10.48175/568

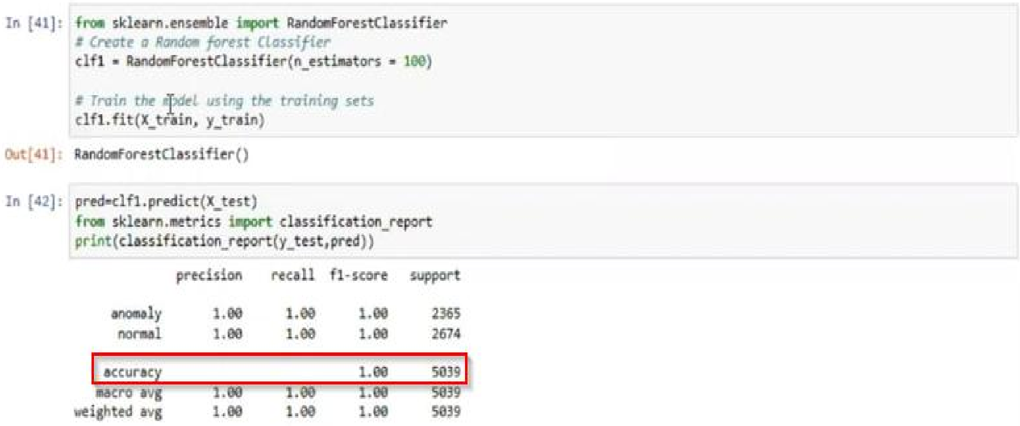
KNN (K-Nearest Neighbor): The KNN algorithm gives a higher accuracy as compared to SVM, thus would be considered a betteroption than implementing the SVM algorithm inthe case of the dataset used.



```
In [43]: from sklearn.neighbors import KNeighborsClassifier
         # Create a Random forest Classifier
         clf =KNeighborsClassifier(n_neighbors=3)

         # Train the model using the training sets
         clf.fit(X_train, y_train)
         pred=clf.predict(X_test)

         print(classification_report(y_test,pred))
         print(X_train.shape)

              precision    recall  f1-score   support

     anomaly       0.99      0.99      0.99      2365
      normal       0.99      0.99      0.99      2674

    accuracy                           0.99      5039
   macro avg       0.99      0.99      0.99      5039
weighted avg       0.99      0.99      0.99      5039

(20153, 41)
```

Figure 2: Accuracy of KNN algorithm

Random Forest: The random forest algorithm stands out to be the ideal algorithm that can be implemented with this dataset as it gives the best result among all the different considered algorithms.

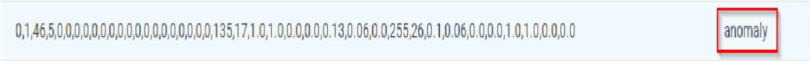


```
In [41]: from sklearn.ensemble import RandomForestClassifier
         # Create a Random forest Classifier
         clf1 = RandomForestClassifier(n_estimators = 100)

         # Train the model using the training sets
         clf1.fit(X_train, y_train)

Out[41]: RandomForestClassifier()

In [42]: pred=clf1.predict(X_test)
         from sklearn.metrics import classification_report
         print(classification_report(y_test,pred))

              precision    recall  f1-score   support

     anomaly       1.00      1.00      1.00      2365
      normal       1.00      1.00      1.00      2674

    accuracy                           1.00      5039
   macro avg       1.00      1.00      1.00      5039
weighted avg       1.00      1.00      1.00      5039
```

Figure 3: Accuracy of Random Forest algorithm

IV. RESULT

When the inputs are provided in the implemented Anomaly Detection System the model detects the class of the traffic i.e. whether it is an normal traffic or is it an anomaly traffic. If the system indicates that the traffic is an anomaly then the admin can take further appropriate actions in order to safeguard the system. The admin can then implement various techniques defined by the specific entity to make sure their system and data is safe. Below is a representation of an input values that fall into anomaly traffic that have been detected by the implemented system.



```
0,1,46,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,135,17,1.0,1.0,0.0,0.0,0.13,0.06,0.0,255,26,0.1,0.06,0.0,0.0,1.0,1.0,0.0,0.0        anomaly
```

Figure 4: Result of the implemented system.

DOI: 10.48175/568

V. CONCLUSION

Thus we have successfully implemented an “Anomaly Detection System” that can prove to be very useful in order to protect valuable user and organizational data by detecting an network Anomaly in early stages by analyzing the network traffic with the help of machine learning algorithms

REFERENCES

[1] Hurley, T.; Perdomo, J.E.; Perez-Pons, “A. HMM-Based Intrusion Detection System for Software Defined Networking. In Proceedings of the 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, USA, 18–20 December 2016; pp. 617–621.

[2] Shone, N.; Ngoc, T.N.; Phai, V.D.; Shi, “Q. A Deep Learning Approach to Network Intrusion Detection”, IEEE Trans. Emerg. Top. Comput. Intell. 2018, 2, 41– 50.

[3] Gomez, J.; Gil, C.; Banos, R.; Marquez, A.L.; Montoya, F.G.; Montoya, M.G. A,“Pareto-based multi-objective evolutionary algorithm for automatic rule generation in network Intrusion detection systems”, Soft Comput. 2013, 17, 255–263.

[4] Sangeetha, S.; Gayathri devi, B.; Ramya, R.; Dharani, M.K.; Sathya, P. Signature Based Semantic Intrusion Detection System on Cloud. In Information Systems Design and Intelligent Applications; Mandal, J.K., Satapathy, S.C., Kumar Sanyal, M., Sarkar, P.P., Mukhopadhyay, A., Eds.; Springer: New Delhi, India, 2015; pp. 657–666.

[5] Dey, S.K.; Rahman, M.M. , “Effects of Machine Learning Approach in Flow-Based Intrusion Detection on Software-Defined Networking”, IEEE 2020

[6] Vipin, Das & Vijaya, Pathak & Sattvik, Sharma &Sreevathsan& MVVNS. Srikanth & Kumar T, Gireesh, “Network Anomaly Detection System Based On Machine Learning Algorithms , International Journal of Computer Science & Information Technology, 2010

[7] Choi, J & Choi, Chang & Ko, Byeongkyu& Choi, D & Kim, “Detecting web based Ddos attack using mapreduce operations in cloud computing environment ” Journal of Internet Services and Information Security, 2013

[8] Baig, Zubair & Baqer, M & Khan, Asad, “A Pattern Recognition Scheme for Distributed Denial of Service (DDoS) Attacks in Wireless Sensor Networks”, 2006

[9] Analyzing Log Files for Post-mortem Intrusion Detection Gamboa, Karen & Monroy, Raúl & Trejo, Luis & Aguirre Bermúdez, Eduardo & Mex-Perera, Carlos. (2012), IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews)

[10] Network Traffic Analysis and Intrusion Detection Using Packet Sniffer Qadeer, Mohammed & Iqbal, Arshad & Zahid, Mohammad & Siddiqui, Misbahur, Communication Software and Networks

DOI: 10.48175/568