Attack Analysis Over Wide Network Transaction Server

Prajwal P[1], Sarath N[2], Dr. Mukesh Krishnan M B[3], Rohith B[4],
Dept. Networking and communications SRM Institute of science and technology Kattankulathur, Tamil Nadu

***Abstract*—Industrial digital transformation has intensified along with the dramatic growth of complex computer attacks throughout recent times. The current intrusion detection systems (IDS) find it challenging to detect contemporary advanced cyber-attack approaches. A Machine Learningbased Intrusion Detection System (IDS) that combines rule-based detection with deep learning techniques exists to accurately identify malicious network activities is proposed for addressing previous system limitations. Users can easily interact with the GUI-based system through Python's Tkinter framework because its design provides a complete interface which enables both novices and experts to work without extensive technical expertise. The system initiates with choosing the dataset before executing multiple preprocessing operations to handle missing data and eliminate unneeded records and extract apt features for model training purposes. The system reaches its critical point through its implementation of two classification techniques. A Decision Tree classifier based on the FURIA inspiration method enables identification of rules which lead to interpretable classifications. The system integrates a Functional Recurrent Neural Network (FRNN) which learns patterns across temporal sequences through its capability to detect advanced intrusion patterns. The joint usage of these models strengthens the detection capability of the system through extensive attack type recognition while increasing accuracy levels. Attack types receive clustering treatment in the solution while extracting statistical results from the dataset. The model performance gets measured through accuracy score with additional evaluations using confusion matrix and classification reports. The system implements Matplotlib and Seaborn visual analytics for generating user-friendly plots which present data from results and system performance. The proposed hybrid approach goes through comparison testing using SVM also known as support vector machines and conventional Decision Tree as baseline classifiers to show its superior characteristics. The solution presents a robust defensive system built with modular components which assures scalability for intrusion detection through data preparation and intelligent classification along with interactive capabilities**

I. INTRODUCTION

Cyber security faces an acute problem in dataset availability within its field. Training needs these essential assets but their acquisition proves difficult then their organization turns into an even bigger challenge. Real system-based data stands as the most useful dataset since it serves as genuine reference material. Release of such highly sensitive data occurs infrequently and is almost never made available to the public domain. The process of data anonymization must be handled carefully by domain experts since the task poses significant difficulties. The substantial size of competition data from capture-the-flag events or hackathons becomes a challenge since the data remains unannotated during the time-consuming curation process. Also the capture originates from network gateway nodes which makes many packets unreadable during encryption while lacking victim or attacker system state data.

Epic 1 The detection of network intruders is made possible through hardware or software systems referred to as IDSs. IDS systems differ between host-based operation and network-based operation as well as the intruder detection approach through misuse detection and anomaly detection. Each IDS system offers different advantages and disadvantages among its many types. A Host based Intrusion Detection System acts as a system monitoring platform that installed on a single machine to detect unauthorized system use either by sending alerts or logging detected activities. The system continuously tracks all internal along with external network occurrences to detect behaviors that may indicate an impending network or system intrusion when someone attempts to penetrate or damage the system. The primary goal of IDS systems along with their fundamental operation exists to secure critical organizational information from unauthorized intruders. IDS stands as a security tool which detects unauthorized

intrusions alongside malicious programs known as attacks in computer systems and networks. IDS systems normally produce notification alerts in addition to showing intrusion points in the system. Detection and identification of attack and normal behavior depend on these common terms. Figure 1 shows anomaly detection process.

Fig 1: Detection parts of intrusion consist of Evaluating and Tracking system operations and user interactions. 1. The assessment includes evaluation of every system configuration element and its vulnerabilities. 2. Analyze the file integrity and the system. 3. The system shows ability to distinguish normal attack signatures 4. Analysis of the anomalous activity patterns. 5. Monitoring the user policy with violations. 6. Network security relies on specifically installed intrusion detection systems which both detect network threats and monitor packet activities. The IDS achieves this through assessment of multiple network and system sources which it uses to evaluate data for security risks.

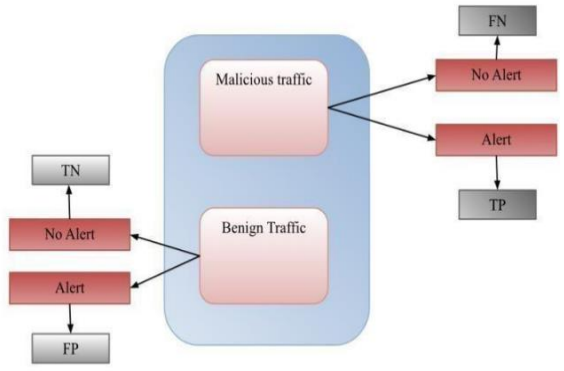


Figure 1 anomaly detection using instances

Epic 2 Information security detection systems operate as either anomaly base or misuse base systems. The anomaly intrusion detection system uses recorded normal behavioral data to detect attacks. A widespread use of this intrusion detection system exists due to its ability to recognize new intrusion types when it compares current real-time traffic with recorded previous normal real- 3-time traffic. From another point of view this system generates the highest number of false positive alarm data by wrongly classifying numerous standard packets as attack packets. The misuse intrusion detection system depends on attack signatures present in a signature repository to perform detection. The detection system does not produce false alarms however it fails to identify novel types of attacks.

Epic 3 Information security detection systems operate as either anomaly base or misuse base systems. The anomaly intrusion detection system uses recorded normal behavioral data to detect attacks. A widespread use of this intrusion detection system exists due to its ability to recognize new intrusion types when it compares current real-time traffic with recorded previous normal realtime traffic. From another point of view this system generates the highest number of false positive alarm data by wrongly classifying numerous standard packets as attack packets. The misuse intrusion detection system depends on attack signatures present in a signature repository to perform detection. The detection system does not produce false alarms however it fails to identify novel types of attacks.

Epic 4 Information security detection systems operate as either anomaly base or misuse base systems. The anomaly intrusion detection system uses recorded normal behavioral data to detect attacks. A widespread use of this intrusion detection system exists due to its ability to recognize new intrusion types when it compares current real-time traffic with recorded previous normal real-time traffic. From another point of view this system generates the highest number of false positive alarm data by wrongly classifying numerous standard packets as attack packets. The misuse intrusion detection system depends on attack signatures present in a signature repository to perform detection. The detection system does not produce false alarms however it fails to identify novel types of attacks.

Epic 5 Our research project removes all possible manual and ad-hoc elements that exist during intrusion detection system construction. We choose a data-oriented method to view intrusion detection as an analytic process of examining data. The search for normal usage patterns in audit data forms anomaly detection whereas misuse detection refers to data-based intrusion pattern encoding followed by matching procedures. The core principle of our system applies data mining strategies to intrusion detection.

II. LITERATURE SURVEY

Manoj Kumar proposed outlier detection as an active field of research within the data mining domain to researchers.[1] This research puts forward DenOD (Density Based Outlier Detection) as a new efficient outlier detection concept for intrusion detection in cloud computing environments through an unsupervised method. The use of unsupervised outlier detection techniques continues to gain significance across various areas including network intrusion detection as well as fault and fraud detection. An outstanding quality of the unsupervised technique is its ability to operate without requiring training data or previous knowledge. Using this method enables the detection of correct and newly emerging attacks even without prior knowledge availability. DenOD implements an IDCC (Intrusion Detection in Cloud Computing) framework featuring the elements of Cloud nodes and IDS (Intrusion Detection System) along with End User. Technical capabilities of this method enable it to discover all types of attacks together with faulty cloud services. The datasets used for intrusion detection serve as essential tools for evaluating machine learning approaches designed for intrusion detection systems according to Shan Suthaharan [2]. Generally, the Intrusion detection datasets exhibit vast sizes with numerous features providing no value and duplicate information. Such negative factors provide erroneous detection of intrusions while requiring increased computational resources during machine learning evaluation. Several researchers have developed procedures to remove redundant information combined with nonimportant attributes from data sources. These reduction techniques decrease data set dimensions while moving the dataset characteristics toward security intrusion patterns that appear in actual network systems. An ellipsoid-based approach is developed to identify anomalies and provides data cleaning for intrusion detection datasets according to this document. The research handles its performance evaluations using freely accessible KDD'99 and NSL-KDD datasets. The NSL-KDD dataset demonstrates an interesting characteristic which manifests through its decreasing behavior. Thework of Yiming Liu [3] during recent times has resulted in numerous proposed approaches 6 to detect intrusions. The present paper introduces a cloud intrusion detection system that uses novel statistical waveforms for classification. The procedure creates waveforms using network connection records over an observation period and then determines probable suspicious characteristics for classification purposes. The system uses particular waveform features to perform intrusion classification. A DARPA Intrusion Detection Data Sets served as the basis for evaluation where our method demonstrated initial feasibility through the testing results. The researchers Anand Sukumar J V, Pranav I, Neetish MM, Jayasree Narayanan [4] developed an intrusion detection system that uses improved genetic k-means algorithm(IGKM) for detecting intrusion types. The paper compares intrusion detection systems that apply kmeans++ and IGKM algorithm by evaluating performance on a thousand-instance subset of kdd99 dataset and the KDD-99 dataset. Experiment results indicate that intrusion detection through IGKM algorithm achieves higher accuracy than k-means++ algorithm. Today's world uses Internet as its dominant channel of operation by populations throughout every continent. The development of science and technology occurred because of this phenomenon. Multiple surveys confirm network intrusion continues to grow annually as it becomes a principal platform for attacks and private information theft in modern times. A computer network becomes the target of unauthorized activities when we refer to network intrusion as such. An effective intrusion detection system needs development due to the present need. The evaluation of these systems depends on either anomaly detection or signature database methods according to Jonathon Ng, Deepti Joshi, Shankar M. Banik [5]. The research incorporates signature database with anomaly detection through data mining techniques. A tool under our solution executes data mining tools against log files to extract suspicious authorized activity patterns. The detection system becomes more effective through the addition of new patterns accumulated from each passing time period. The system detected both brute force password cracking attempts along with Denial-of-Service (DoS) attacks in the Ubuntu operating system platform. We introduce the most common attacks which a detection system must recognize in this following part. For this work study we examined the DoS Attacks as one of the presented attack types. The attacks serve to

deprive authorized users of system access. The author Ajith Abraham of [7] expanded the soft computing techniques and studied their relative uses in detail. A system called Intrusion Detection System (IDS) operates as a program 7 that examines system activities during executions to detect signs of computer misuse. The research examines three fuzzy rule based classifiers for IDS with decision trees, support vector machines and linear genetic programming performance assessment included. Soft Computing (SC) based IDS (SCIDS) functions as a set of interconnected different classifiers to establish advanced and lightweight (heavy weight) IDS. The empirical evidence demonstrates that the intrusion detection role of SC methodology could become fundamental. The paper of Ahmed Ahmim [8] presents a new hierarchical intrusion detection system that combines REP Tree with JRip algorithm and Forest PA as rule-based and decision tree classifiers. The first and second method accept data features as input while determining whether network traffic belongs to Attack or Benign. The third classifier integrates both original data set features with combined first and second classifier outputs. Study results on the proposed IDS reveal advanced performance metrics when using the CICIDS2017 dataset for accuracy detection rate and false alarm rate and time overhead than contemporary IDS methods. Herv´e Debar notes that intrusion-detection systems work to discover computer system and network attacks as well as attacks on information systems in general terms. The task of maintaining information systems in a secure state which withstands any age of operation proves challenging due to their difficulty to prove absolute security at the start. Information systems do not always support the implementation of complete security measures because of operating or legacy system restrictions. Information systems need intrusion-detection systems to monitor their usage in order to identify unsafe system states. Intrusion-detection systems track both internal members' active security breaches and external persistent security violations of information systems. The primary part of this extended work presents an introduction to intrusion-detection system taxonomies which will be followed by the second part. The research by Chen Yan [10] introduces a new intelligent intrusion detection system model to address the typical issue of incorrect detection rates in intrusion systems. The proposed model implements neural network weight optimization through genetic algorithm because of its global superiority and nerve local properties. The experiment data reveals that employing intelligent methods builds up the efficiency of intrusion detection systems. This paper discusses an intelligent intrusion detection model that addresses the missing report rate alongside false alert rate which commonly occur in intrusion detection systems. The model utilizes genetic algorithm because of its global optimization power alongside the local capabilities of nerve 8 networks to optimize neural weights. The experimental data indicates that employing an intelligent method boosts the detection capacity of intrusions.

III. METHODOLOGY

A. Data Acquisition

Our experience with developing a distributed intrusion detection system allows us to analyses both intrusion detection data requirements together with the system constraints which stem from built-in data collection features in standard operating systems. The collection of data should occur directly from host operations instead of relying on data found in audit trails or network packets when a system delivers its intrusion detection function. The source code of operating system applications and its underlying operating system requires built-in monitoring mechanisms to enable efficient complete and reliable data collection processes. KDD dataset receives its data collection from the original source. The Data Acquisition (SCADA) systems operate without a security protection method that surpasses various intrusion techniques or stops exposure of data when other applications specifically Intrusion Detection System (IDS) manage the data.

B. Data Preprocessing

Processing data stands as a fundamental requirement during the data mining operational sequence. Data mining projects along with machine learning applications require special attention to the principle "garbage in, garbage out." Data-preprocessing techniques include cleaning and integration as well as transformation and reduction while handling the frequent occurrence of uncontrolled data collection issues which produce out-of-range values and

impossible data combinations together with missing values and more. This research details the full description of preprocessing techniques adopted specifically for mining data purposes. An information system contains data objectsthat researchers also describe through various names get their description through multiple features which represent essential characteristics of objects such as physical mass measurements and event timing information etc. These features also have alternative names including variables, characteristics, fields, attributes, or dimensions.

Data Quality Assessment The processing of machine learning data usually requires significant effort because unreliable data from various inconsistent sources occupies more than fifty percent of our time. Nobody can reasonably anticipate perfect data quality. Three different types of problems can appear during data collection due to human mistakes and inadequate measuring equipment and faulty procedures. Under the following topics we will examine several issues along with correction techniques

Missing values: Most datasets include missing data points in their records. All missing values require attention no matter when they occurred or why during the data collection process. Eliminate rows with the missing data Simple and sometimes effective strategy. If most values for a single feature are missing the feature itself may qualify for elimination. When acceptable amounts of values are missing we can execute basic interpolation systems to replace those empty values. The most frequent way to handle missing data consists of substituting it with the mean or median or mode value from that specific feature.

Inconsistent values Within the system data exists with inconsistent values which the system understands. It is likely that each one of us has encountered this problem in the past. The 'destination port field displays 'IP address'. Either human mistakes or poor reading during handwritten document data capture could be responsible for this situation. The assessment of data requires users to determine the correct data types for all features as well as validate that they match for the entire set of data objects.

Duplicate values: Duplicated data objects exist within the dimension of a particular dataset. One person can fill in the form multiple times to lead to duplicate data objects. The operation to deal with duplicates is called reduplication according to industry terminology. The majority of duplicate entries receive elimination procedures because they could present unintended biases to specific data objects throughout machine learning algorithm work

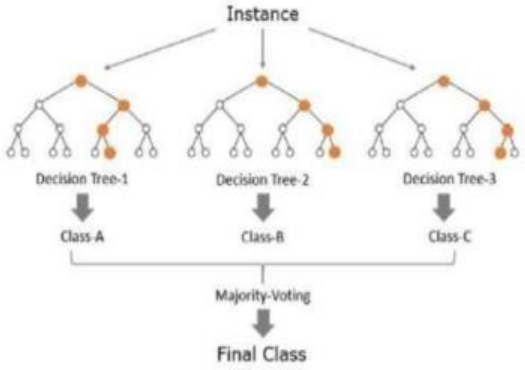
C. . Gaussian Feature Extraction

Organizations handle datasets consisting of hundreds (or sometimes thousands) of features in their standard operations. A dataset with equal or greater number of features in relation to observations will likely cause a Machine Learning model to suffer from over fitting. Two solutions exist for preventing this problem through either regularization techniques or dimensionality reduction methods (Feature Extraction). The data dimension within Machine Learning corresponds to the total number of variables that constitute the database representation. The application of Regularization methods would decrease overfitting risks yet Feature Extraction techniques bring alternative benefits to the table such as: Accuracy improvements. Over fitting risk reduction. Speed up in training. Improved Data Visualization. The model obtains enhanced explain ability. According to Bayes theory the best Gaussian feature extraction requires computing the matrix square root operation on covariance matrix inverse $\Sigma - 1 / 2$. A test of the proposed algorithm was conducted to extract Gaussian features from Gaussian data having three distinct classes and existing in three dimensions. The machine learning community has steadily increased its interest in Gaussian Process for Machine Learning (GPML). This document presents a proposed method to execute GPML for image classification procedures. This algorithm depends on basic image features which extraction occurs swiftly and efficiently. A test of the proposed algorithm happens on Caltech 256 data sets while performing a comparison with Least Squares Support Vector Machines (LSSVM) The set of classes includes L specific elements to which our patterns belong while n $x \in \Re$ represents a pattern vector whose mixture distribution follows p(x). A system that includes U U prior probabilities P(ωi) ,i = 1,..., L together with L conditional probability densities p(x |ωi) ,i = 1,...L and L posterior probabilities P(ωi x) , i =1,...L is assumed. A pattern belongs to ωi based on Bayes classification rule when the conditions P P i j L and j i (ωi | x) > (ω j | x) , , = 1,..., ≠ are satisfied. Ln P(| x) Ln p(x |) Ln P( ) ln p(x) represents the Bayes theorem when calculating the posterior probability term P ωi (x).

Under equal prior probability conditions, the classification of incoming data depends exclusively on the value ranging from 1 to L. The 19 sufficient information required for identifying Gaussian data minimizes Bayes error through the feature which transforms Gaussian distribution to a quadratic function (x) i f . Online applications need to determine values of −1 2 Σ and mi which are usually unknown quantities. A rule must be developed to adaptively determine these values before (x) i f can be calculated. The following section introduces new approaches combined with network solutions for adaptive −1 2 Σ calculation.

D. Classification

The module implements Random Forest together with XGBoosting as attack classification methods. The Random Forest algorithm has been selected for its exceptional accuracy and its ability to work with imbalanced datasets since it achieved a detection accuracy of 90.6%. XGBoost operates efficiently to handle difficult data patterns along with its processing speed and delivers an 84% accurate result. The predictive algorithm combines Random Forest with XGBoosting technology which improves accuracy levels by suppressing false outputs and negative results. The evaluation system uses precision rate alongside recall rate and F1 score as measurement standards. The applied method strikes a suitable equilibrium between detection rate and real-time speed when identifying intrusions in real time



.

E. Intrusion Prediction

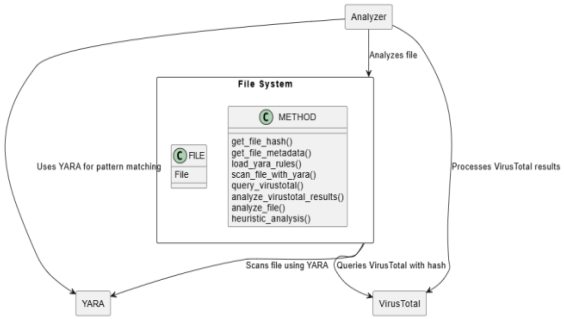
The research adopts anomaly detection techniques and signature database management mechanisms based on the data mining process. A tool through our system enables data mining tools to scan log files for identifying unauthorized patterns in the data. The system owner confirms suspected attacks which then get archived within the signature database. The detection capability of our tool improves automatically as new patterns emerge during its operations. The latest version of our tool combines DoS attack detection with brute force 20 password cracking by using its repeated log entry clustering ability. A smart intrusion detection framework receives evaluation in the article because it tackles detection system false positives and missing alerts. The weights of a neural network are optimized using genetic algorithm which applies its global superiority property in combination with nerve's local characteristics. The intelligent approach demonstrates capability to boost the efficiency levels of intrusion detection systems according to experimental findings. The classification system tries the process of classifying: The correctly classified instances Incorrectly classified as instances. Error rate. Precision. Recall. ROC. Accuracy Within the given instances accuracy and precision of the system the difference of variations between the FRNN and FURIA is well known.n.

F. Deployment and Monitoring

Once the system is completely tested and developed, it is deployed in a production environment for constant monitoring of the digital infrastructure of the organization

G. Documentation and Reporting

Complete documentation is required to share the findings, performance metrics, and overall effectiveness of the system

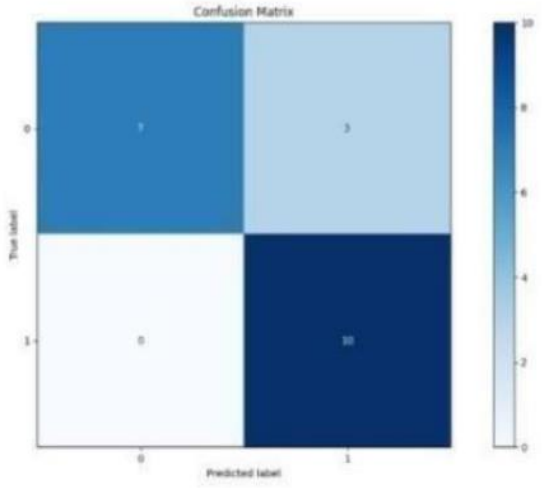


IV. IMPLEMENTATION

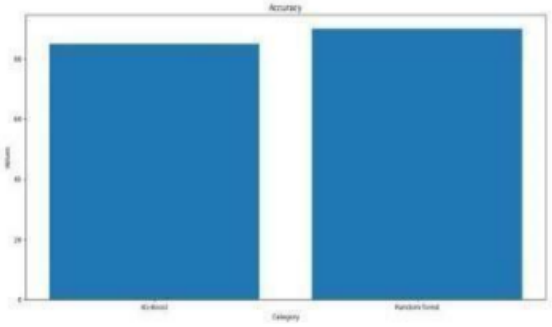
classification model performance evaluation consists of using the confusion matrix to analyze actual and predicted outcomes. A confusion matrix presents a

complete accuracy breakdown through its presentation of correct predictions together with specific error instances. The matrix classification matrix includes four essential elements between True Positive (TP) with True Negative (TN) while including False Positive (FP) and False Negative (FN). The predictions identified as TP and TN indicated correct results but the model incorrectly identified cases as both FP and FN. The confusion matrix helps evaluate model performance by providing analysis of its components while it measures accuracy and precision and recall alongside calculating the F1 score.. • The prediction matches both the true output being positive while the actual output reveals the same positive result. • Two conditions match where the prediction classifies as negative and reality confirms the same negative output. • A positive prediction corresponds to a false positive case because the actual output shows negativity. • A model identifies negative output but true results reveal positive status under this condition. Accuracy: The accuracy metric computes correct prediction count while considering the total number of model predictions. General model performance assessment is possible through accuracy but the metric can deliver false results when working with class-imbalanced datasets. Accuracy=TP+TNTP+TN+FP+FN 22 Precision: In precision calculation the accuracy percentage reflects which predicted positive cases actually prove to be positive. positive. The use of precision evaluation becomes essential for systems that need to minimize false positive results such as instances of spam analysis. Precision=TpTP+FP Recall: Recall measures how well the model identifies all actual positive cases. It is important when it is critical not to miss positive cases, such as in medical diagnostics. Recall=TPTP+FN F1-Score: The calculation of harmonic mean between precision and recall produces the F1- Score metric. The F1-Score provides optimal results in cases where accurate detection of both false positives and negatives matters and occurs with unequal data distributions. F1-Score=2.Precision. RecallPrecision+Recall Specificity: The model determines its ability to label negative cases correctly through the specificity measurement. Model specificity receives another name as True Negative Rate and works significantly for negative case identification. Specificity=TNTN+FP

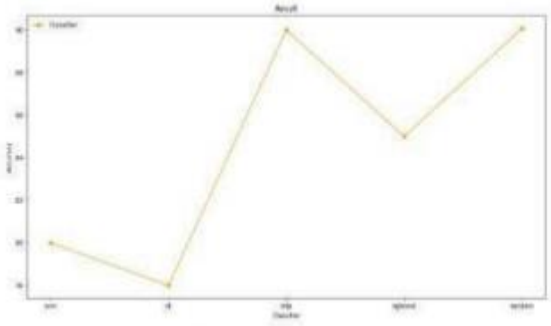
XGBoost and Random Forest algorithm performance evaluation relies on the confusion matrix to display TP, FP, FN and TN values. Attack identification accuracy becomes possible through these algorithms because they show high detection rates together with minimum false negative occurrences. Random Forest creates a superior precision to recall balance that minimizes false alerts while detecting suspicious activities well.



A bar graph evaluates the performance results from Random Forest against XGBoost based on accuracy and precision and recall and F1 score metrics. When using Random Forest yields superior accuracy along with recall performance when compared to XGBoost because of improved attack type generalization capabilities. While XGBoost delivers superior precision scores than the other approaches it fails to identify some attack situations



The proposed models demonstrate their superiority against five algorithms including NLP and Decision Tree and SVM and Random Forest and XGBoost as shown by this graph. Retrieving insights from the Random Forest method produces stable performance throughout all evaluation metrics with XGBoost demonstrating equivalent accuracy levels though slightly worse recall scores. Ensemble techniques at an advanced level prove their superiority in attack detection effectiveness The presented time-based graph demonstrates processing durations between training and prediction steps of five different decision systems. The speed of small-dataset execution lies with NLP and Decision Tree models yet Random Forest and XGBoost achieve optimal speed and accuracy for larger data collections. Possible trade-off performance is achieved through Random Forest yet XGBoost requires a longer execution time because of its complex gradient boosting method



V. CHALLENGES AND INNOVATION

The existing implementation generates positive results yet various opportunities exist to enhance it for practical deployment. Complex system configuration along with management stand as one of the main limitations for these systems. User and organizational challenges emerge because the models need both initial setup adjustments and continuous calibration even though they exhibit good performance. The shortage of trained security professionals exists at large in operational settings including industrial and governmental infrastructures. The management of intrusion detection systems frequently happens under junior analysts or system administrators although they normally lack proper qualifications to read model outputs and modify configurations. The skills deficit affects the potential performance of otherwise excellent systems. Future modifications of the system should include features for automated alert interpretation together with self-learning abilities and natural language reporting to boost decision-making abilities of inexperienced users. The IDS performs better when connected to a Security Information and Event Management (SIEM) platform since it enables incident correlation analysis as well as real-time threat intelligence sharing and improved response coordination abilities. Development of live packet-level observation rests on combining the detection framework with Snort or Suricata tools together with model-specific online learning adaptations. The system can dynamically adapt through new attack patterns by avoiding the need for complete retraining processes. The addition of cloud deployment abilities together with mobile dashboard support would make the system flexible enough to run in smart city infrastructure and remote industrial IoT areas as well as vital public sector networks. In summary, while the current system offers a robust foundation for intrusion detection in WSNs, future enhancements must focus on automation, usability, scalability, and integration with broader cybersecurity ecosystems to maximize its impact and usability in real-world environments.

VI. ACKNOWLEDGMENT

We express our humble gratitude to Dr. C. Muthamizhchelvan, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support. We extend our sincere thanks to Dean-CET, SRM Institute of Science and Technology, Dr.T.V. Gopal, for his invaluable support. We wish to thank Dr. Revathi Venkataraman, Professor & Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work. We encompass our sincere thanks to Dr. M. Pushpalatha, Professor and Associate Chairperson, School of Computing and Dr. C. Lakshmi, Professor and Associate Chairperson, School of Computing, SRM Institute of Science and Technology, for their invaluable support. We are incredibly grateful to our Head of the Department Dr. M. Lakshmi, Professor and Head, Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the

project work. We want to convey our thanks to our Project Coordinator, Dr. G. Suseela, Associate Professor, Panel Head Dr. G. Sujatha and Panel Members Dr. M.B Mukesh Krishnan, Professor, Dr. T. Balachander, Assistant Professor, Dr. R. Lakshminarayanan Assistant Professor Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, for their inputs during the project reviews and support. We register our immeasurable thanks to our Faculty Advisor, Dr. G. Abinaya, Dr. D. Saveetha Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, for leading and helping us to complete our course. Our inexpressible respect and thanks to our guide, Dr. M.B Mukesh Krishnan, Professor, Department of Networking and Communications, SRM Institute of Science and Technology, for providing us with an opportunity to pursue our project under her mentorship. She provided us with the freedom and support to explore the research topics of our interest. Her passion for solving problems and making a difference in the world has always been inspiring. We sincerely thank the Networking and Communications department staff and students, SRM Institute of Science and Technology, for their help during our project. Finally, we would like to thank parents, family members, and friends for their unconditional love, constant support, and encouragement.

REFERENCES

[1] Singh, R. K., & Ramanujam, T. (2009). Intrusion Detection System Using Advanced Honeypots. International Journal of Computer Science and Information Security (IJCSIS), Vol. 2, No. 1.

[2] Alqahtani, A. S., Altammami, O. A., & Haq, M. A. (2024). A Comprehensive Analysis of Network Security Attack Classification using Machine Learning Algorithms. International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 15, No. 4.

[3] Malav, S., Avinash, M. S., Satish, N. S., & Sandeep, S. C. Network Security Using IDS, IPS & Honeypot. International Journal of Recent Research in Mathematics Computer Science and Information Technology.

[4] Yeldi, S., Gupta, S., Ganacharya, T., Doshi, S., & Bahirat, D. Enhancing Network Intrusion Detection System using Honeypot. IEEE Xplore.

[5] Arifianto, R. M., Sukarno, P., & Jadied, E. M. (2018). An SSH Honeypot Architecture Using Port Knocking and Intrusion Detection System. 6th International Conference on Information and Communication Technology (ICoICT).

[6] Jawale, S., Mehta, R., Mahalingam, V., & Mehta, N. (2012). Intrusion Detection System using Virtual Honeypots. International Journal of Engineering Research and Applications (IJERA), National Conference on Emerging Trends in Engineering & Technology.

[7] Edwin, G. K., Ewards, S. V., Kathrine, G. J. W., Palmer, G. M., Bertia, A., & Vijay, S. J. (2022). Honeypot based Intrusion Detection System for Cyber Physical Systems. Proceedings of the International Conference on Augmented Intelligence and Sustainable Systems (ICAISS).

[8] Dongxia, L., & Yongbo, Z. (2012). An Intrusion Detection System Based on Honeypot Technology. International Conference on Computer Science and Electronics Engineering.

[9] Jogdand, P., & Padiya, P. (2016). Survey of Different IDS Using Honeytoken-Based Techniques to Mitigate Cyber Threats. International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT).

[10] Kondra, J. R., Bharti, S. K., Mishra, S. K., & Babu, K. S. (2016). Honeypot Based Intrusion Detection System: A Performance Analysis. International Conference on Computing for Sustainable Global Development (INDIACom).

[11] Yang, X., Yuan, J., Yang, H., Kong, Y., Zhang, H., & Zhao, J. (2023). A Highly Interactive Honeypot-Based Approach to Network Threat Management. Future Internet, Academic Editor: Georgios Kambourakis.

[12] Zielinski, D., & Kholidy, H. A. An Analysis of Honeypots and Their Impact as a Cyber Deception Tactic. State University of New York (SUNY) Polytechnic Institute. Available at: ResearchGate.

[13] Yang, Y., & Mi, J. (2010). Design and Implementation of Distributed Intrusion Detection System Based on Honeypot. 2nd International Conference on Computer Engineering and Technology.

[14] Gajjar, H., & Malek, Z. Network Intrusion Detection System Using Honeypot in Cloud Environment. International Journal of Intelligent Systems and Applications in Engineering.

[15] Jeremiah, J. Intrusion Detection System to Enhance Network Security Using Raspberry PI Honeypot in Kali Linux. Limkokwing University of Creative Technology.

[16] Baykara, M., & Das, R. (2015). A Survey on Honeypot Technologies Used in Intrusion Detection Systems. 16th ISERD International Conference, Prague, Czech Republic.
[17] Tripathi, S., & Kumar, R. (2018). Raspberry Pi as an Intrusion Detection System, a Honeypot and a Packet Analyzer. International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS).
[18] Javier Galbally, Iwen Coisel, Ignacio Sanchez. A New Multimodal Approach for Password Strength Estimation—Part II: Experimental Evaluation. IEEE Transactions on Information Forensics and Security, July 2017.