Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777851

A NOVEL RULE-BASED INTRUSION DETECTION FRAMEWORK FOR SECURE WIRELESS SENSOR NETWORKS

Manu Devi[1], P. Nandal[2], Harkesh Sehrawat[3]

[1] Department of Computer Science and Engineering, University Institute of Engineering and Technology, Maharshi Dayanand University, Haryana, India. manughanghas26@gmail.com.

[2] Department of Computer Science and Engineering, Maharaja Surajmal Institute of Technology, GGSIP University, Delhi, India priyankanandal@msit.in.

[3] Department of Computer Science and Engineering, University Institute of Engineering and Technology, Maharshi Dayanand University, Haryana, India, sehrawat_harkesh@yahoo.com.

ABSTRACT

In order to safeguard "Wireless Sensor Networks (WSNs)," this research suggests a new rule-based intrusion detection architecture that includes a number of security techniques. The framework employs a set of rules generated by the Random Forest algorithm, which outperforms other machine learning algorithms in terms of detection accuracy, false positive rate, and overhead. The generated rules are based on features extracted from WSNs data, including packet size, energy consumption, and hop count. The framework's performance is evaluated using a set of metrics, demonstrating its effectiveness in identifying numerous attack kinds. Combining rule-based systems and machine learning approaches in this proposed framework can improve the accuracy and efficiency of intrusion detection mechanisms for secure WSNs. The proposed framework can be a valuable addition to the existing intrusion detection mechanisms for secure WSNs.

Keywords: Intrusion Detection System, Wireless Sensor Networks, Security Attacks, Machine Learning,Base Station,Security threat.

1. INTRODUCTION

Security software, known as an "Intrusion Detection System (IDS)is intended to automatically notify administrators when something or someone tries to breach the cybersecurity of a computer or network. IDSs can be configured to monitor for a wide range of security threats, including viruses, worms, unauthorised access, and other types of malicious activity. An IDS in a wireless sensor network (WSN) is a network security device that scans network traffic for irregularities or violations of established rules and warns users when it does so. The aim of an Intrusion Detection System in a WSN is to locate and report any unauthorised or malicious activity that could compromise the security or integrity of the network. There are several challenges in implementing IDS in WSNs, including the limited computational and storage resources of the sensor nodes, the constrained bandwidth and energy of wireless communication, and the dynamic and uncertain nature of the wireless environment. This research work has demonstrated the potential of a rule-based IDS for securing WSNs.

1.1 Literature Survey

Rajendran (2019) proposed the most often referenced rule-based intrusion detection technique for WSNs to identify a variety of attacks in various tiers. A failure was raised, and the counter

was increased by one if the text analysis violated any of the rule tests in the rule application phase, which applied pre-defined rules to the saved data from the previous phase. This plan contains three phases: data collecting, rule application, and data analysis. Comparing the number of successes to failures during the intrusion detection phase, more failures led to the activation of intrusion alarms. In addition, Khan & Hermann (2019) introduced an IDS that used information from nearby nodes to identify attacks that depleted resources and impersonated other nodes. Each node can build a statistical profile of its neighbour using the acquired power rate and arrival packet rate. For a conventional WSN application with numerous dynamic attacks, this method cannot be generalised.

However, Shang et al. (2019) developed a Cumulative Sum algorithm (CuSum)-based WSN attack detection scheme. The CuSum algorithm detects changes in incoming and outgoing packets and collisions in this scheme. At least one monitoring node has been chosen for each sensor node. Since the monitor node was considered a sensor node, it can fail easily. The algorithm has also consumed power in a normal monitor node. The research conducted by Mehetre et al. (2019) developed a lightweight WSN selective forwarding and blackhole detection scheme. Their scheme involves nodes monitoring their neighbourhood and communicating to determine if an intrusion has occurred. The plan is tested with an actual WSN deployment. In the plan, a detecting agent node has had its compute load distributed through neighbour monitoring. Nonetheless, during voting collaboration, nodes were sending more messages, increasing communication overhead and depleting power quickly.

Jatii & Sontif (2019) discussed a more specific sinkhole attack intrusion detection scheme. "Local Packet Monitoring," "Local Detection Engine," "Cooperative Detection Engine," and "Local Response Model" made up this scheme. TinyOS with MinRoute protocol implemented the proposed scheme. The suggested plan satisfies IDS distribution requirements in a sizable, autonomous environment like Wireless Sensor Network. It has been difficult to share relevant attack detection information across nodes due to the communication overhead. Further, Tan et al. (2019) suggested partitioning the sensor network into many algorithms and grouping nearby sensors with similar sensing capabilities. This scheme considered multiple sensor node attributes to improve detection, according to the authors. The clustering and statistical distribution methods computing cost was the primary issue because the common sensor node has constrained resources. This approach utilises a variety of network traffic properties, which has various advantages, including detecting universality.

The rule-based intrusion detection framework was efficient, according to Awotunde et al. (2021). The uncertainty in deciding how many monitoring nodes should be used for detection, how to select them, and precisely how to encompass the entire network was this scheme's fundamental flaw. The scheme only covers certain types of attacks, so what happens if new ones appear? Any intrusion detection scheme should consider these drawbacks in future.

1.2 IDS Leverage Machine Learning Techniques

Based on machine learning anomaly based on prior findings, IDS generates an implicit or explicit model of the examined patterns that is updated on a regular basis to improve system performance.(2022). An online hyper grid "KNN model" was presented by Poornima & Paramasivan (2020) to identify random errors and cyberattacks. This model simplifies computation and communication by changing anomaly detection from hypersphere to

hypercube. A shared decision tree approach was created by Gebremariam et al. in 2021 to identify "sinkhole attacks" in Wireless Sensor Networks. NS3 generates normal and attack traffic for the author. IDS has central and local agents. Threshold values are used by local agents on each sensor node in WSNs to find sensor incursions and notify them of the base station (BS). Traffic is classified as normal or attacked the base station (BS) central agent module. In order to identify intrusions in the NSL-KDD dataset, Yang et al. (2019) employed "spectral clustering (SC)" and "deep neural networks (DNN)." Each cluster classifies normal and attack traffic with one DNN and aggregates all DNN outputs. A widespread WSN intrusion detection model employing "game theory and fuzzy Q-learning" was proposed by Thivakaran and Sakthivel in 2019. The detection and defence of "sink node" and "BS" intrusions were performed using fuzzy Q-learning and game theory.

Ifzarne et al. (2021) used an NS2 network simulator to create the new WSN IDS dataset (WSN-DS) with five cases: "normal," "blackhole attack," "flooding attack," "scheduling attack," and "gray hole attack." Authors detected WSN-DS attacks with ANN. Godala & Vaddella (2020) used "fuzzy C-Means (FCM)," "one-class SVM," and "sliding window" to detect blackhole and flooding attacks. Machine learning methods require high computing resources during anomaly detection training and testing, which harms resource-constrained sensor nodes. Machine learning algorithms detect intrusions accurately.

2. MATERIAL AND METHOD

Figure 1 depicts a methodology for creating ML-based prediction models for intrusion detection. Before creating the ML models for intrusion detection, data pre-processing is a crucial step. By handling the categorical values and missing values, the data set is transformed into one that is suitable for ML models. If the data set contains a large number of attributes, feature selection is also required as part of data preparation. Once the data is prepared, various ML models are created, and a suitable ML model is found by reviewing the outcomes. To identify various attacks, rules are generated using the chosen model.

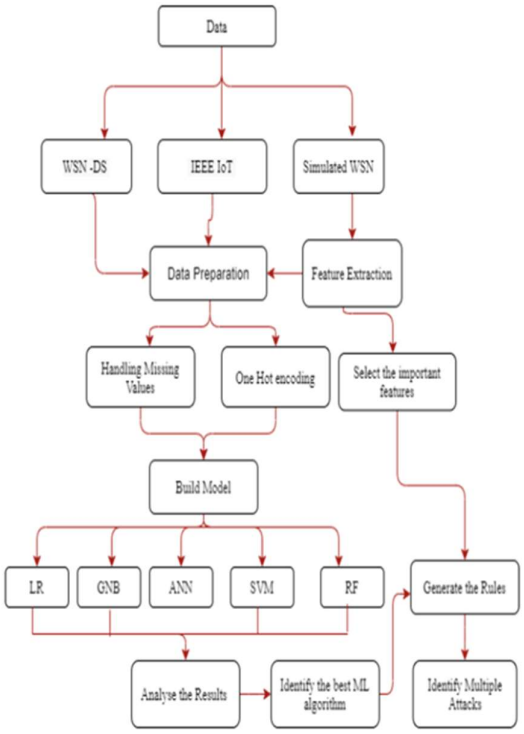

Figure. 1 Methodology for rule-based IDS

2.1 Dataset

CIC-IDS 2017 is the Canadian Institute for CyberSecurity's July 2017 cyber data collection (Stiawan, 2020). One of the newest intrusion detection datasets, CIC-IDS2017, closely matches real-world data. It has recent, mild attacks. 2,830,743 records are in 8 files. Each record has 80+ attribute feature dimensions and tags. It also presents CIC FlowMeter's network traffic analysis results, which are free on the Canadian Institute for Cybersecurity's website. Time-stamped data with attack vectors, protocols, IP addresses, and ports are output (CSV files). CIC-IDS2017 has three attribute groups. Space, time, and content features are in the first, second, and third sections. CIC-IDS2017 covers all eleven criteria, including updated attacks like Port scan, Botnet, Infiltration, SQL Injection, XSS, Brute Force, DoS, and DDoS.

2.2 Pre-processing of Data

Pre-processing is required to get the data sets under discussion and the simulated data ready for usage with ML models because they are raw data. Handling missing values and one hot encoding are used to perform the pre-processing. Due to the fact that the features are protocol-specific, the technique used is to replace the missing values with zero. Using a single hot encoding technique, a new column is added for each value of the categorical attribute.

Before creating analytical models using the data, data exploration can help to understand the data and to determine the relationship between the variables. This analysis can be done by combining two or more attributes or by analysing each attribute separately.

2.3 Models for machine learning

To test the intrusion detection on data sets, machine learning models are constructed using "Logistic Regression (LR)," "Gaussian Naive Bayes (GNB)," "Random Forest (RF)," "Decision Tree (DT)," and "Linear Discriminant Analysis(LDA)" classifiers. Pre-processing

the data involves handling categorical values and missing values. Training data make up 80% of the data, whereas testing data make up 20%. By looking at the ML models' evaluation metrics, a suitable model is found.

3. RESULT AND DISCUSSION

The results of the supervised machine learning algorithms LR, GNB, DT, LDA, and RF on the CIC-IDS2017 dataset are discussed in this section. The generated rules are also used to identify multiple attacks. These algorithms' effectiveness is evaluated using a variety of metrics. These metrics are based on the below-displayed confusion matrix:

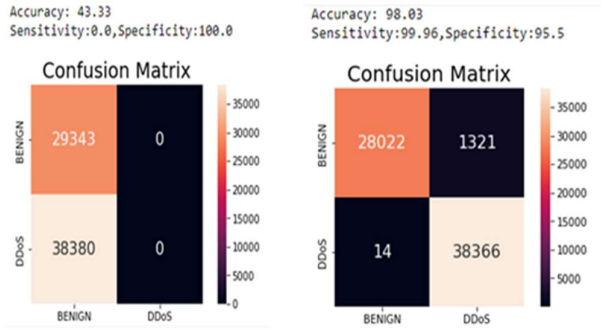

Figure 3. Logistic regression and linear discriminant analysis

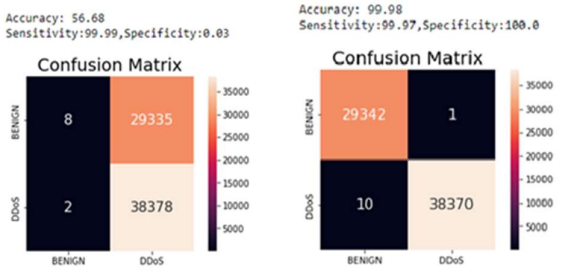

Figure 4. Naïve Bayes and Random Forest

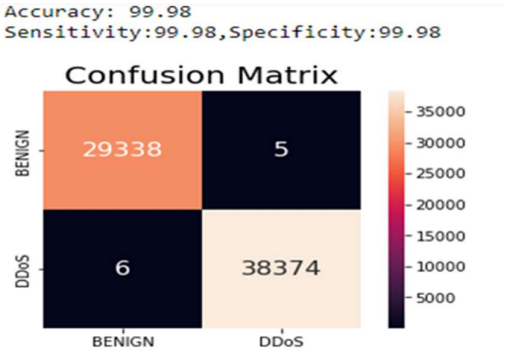

Figure 5. Decision Tree

While random forest had the highest detection accuracy at 99.98%, logistic regression had the lowest accuracy at 43.33% due to its incorrect classification of 38,380 instances. At 99.97%, the decision tree came in second, and at 98.03% for, the linear discriminant analysis. The accuracy rate of the naive Bayes was 56.68%. The sensitivity findings for all considered data instances are then discussed. Logistic regression displays the lowest sensitivity, which is zero,

while Nave Bayes has the highest sensitivity of 99.99%. Random forest and decision tree sensitivity is 99.97%. The sensitivity of linear discriminant analysis was 99.96%. In terms of the specificity results for each classifier, random forest and logistic regression had the highest specificity of 100%, while naive Bayes had the lowest precision. Table 1 summarises the facts above.

Table 1: Comparison of the classification models utilised in this investigation.

Algorithm	Accuracy	Sensitivity	Specificity
Logistic Regression	43.33	0.0	100.0
Linear Discriminant Analysis	98.03	99.96	95.5
Naive Bayes	56.68	99.99	0.03
Random Forest	99.98	99.97	100.0
Decision Tree	99.97	99.97	99.98

3.1 Rule Generation for Multiple Attacks

The Random Forest is the top intrusion detection model, according to the comparative analysis in the previous section. As a result, the rules are generated using the random forest's model parameters. The IDS agents are equipped with these rules for monitoring and spotting various attacks. On the base station, the RF-based rule-generating model is put into action. Regular updates are made to the rules. Algorithm 1 outlines the steps for using Random Forest to generate rules.

Algorithm 1 Random Forest Rule Generation (RFRG)

```
procedure GENERATE-RULE(Train – set – TrDs)
Input: Training set TrDs = (X1, y1) (X2, y2), = (X3, y3)….. (Xn, yn) Result: Rule set RLs = Rl1, Rl2, ……Rlp
    DtreeN=Number of decision trees to construct in random forest
    for i=1 to DtreeN do
        Bstrap = BootStrapSampling(training set TrDs)        ▷ Bstrap is subset from TrDs without replacement
        TrDi = Decision tree using Bstrap
        Rli = All the rules generated by TrDi
        Rall = R U Rli ▷ Rall is the set containing all the rules generated
    end for
    OptRules =ϕ                          ▷ OptRules : Set with optimized rules
    for each sample k in the test set do
        PVk = Prediction using majority voting        ▷ Prediction using ensemble approach
        IF PVk== yi ▷ Correct Prediction using Ensemble majority voting
        CrrPEM++;          ▷ Count of correct prediction using Ensemble majority voting
        for each Rule i in Rall do
            PRki = Prediction using Rule i ▷ Prediction for sample k using Rule i
            if PRji==yi then              ▷ Correct Prediction using Rule i
                CRl++              ▷ Count of correct prediction using Rule i
            end if
        end for
        if CRl > 0.5*CrrPEM then
            OptRules = OptRules U Rli   ▷ OptRules: Optimized Rule Set
        end if
    end for
    return
end procedure
```

The labelled WSN intrusion data is divided into training and test data (designated as D-Training and D-Test, respectively) in the algorithm. Random Forest, which creates dtN numbers of decision trees, is built using D-Training. The set Ri contains the rules from the decision tree DTi. Every rule is combined and saved in set R. Finding the best rules for intrusion detection comes after the Rule-set has been built. The test set TrDs-test is used to find the best rules. Three steps are employed for that. Through using the ensemble approach of the random forest and majority voting, predictions are first created for each sample in the test set. The CrrPEM count is increased for an accurate prediction. In the second step, Rulei is used to produce predictions for each sample in the test set. After verifying that the prediction is accurate, the CRl count is increased. Rli is added to the Optimal Rule-set after being evaluated for optimality in the third step, under the condition that the correct prediction count using Rulei is at least 50% better than the predictions using the ensemble approach (OptRules). The base station periodically executes algorithm 1 to generate and dynamically update rules. Following that, the rules are put into action to look for malicious node behaviour at the base station. The IDS agent uses the generated rules to categorise intrusions as malicious communication while in the monitoring state.

4. CONCLUSION

Wireless Sensor Networks (WSNs) are widely used for monitoring and data gathering, but their wireless nature makes them vulnerable to security threats. To address these security challenges, a rule-based intrusion detection framework has been proposed as a solution. This framework uses pre-defined security rules to identify and respond to potential security threats in real time. This research paper has demonstrated the potential of a rule-based IDS for securing WSNs. Further research and implementation can help to refine and improve this framework, making it an even more powerful tool for protecting WSNs from security threats.

REFERENCES

1. Rajendran, R., Santhosh Kumar, S. V. N., Palanichamy, Y., & Arputharaj, K. "Detection of DoS attacks in cloud networks using intelligent rule based classification system". Cluster Computing, vol.22, pp. 423-434, 2019.
2. Mehetre, D. C., Roslin, S. E., & Wagh, S. J. "Detection and prevention of black hole and selective forwarding attack in clustered WSN with Active Trust". Cluster Computing, Vol. 22, PP. 1313-1328, 2019.
3. Khan, Z. A., & Herrmann, P. "Recent advancements in intrusion detection systems for the internet of things", Security and Communication Networks, 2019.
4. Shang, F., Zhou, D., Li, C., Ye, H., & Zhao, Y. "Research on the intrusion detection model based on improved cumulative summation and evidence theory for wireless sensor network", Photonic Network Communications, no.37, 212-223, 2019.
5. Jatti, S. A. V., & Sontif, V. K ". Intrusion detection systems", International Journal of Recent Technology and Engineering, vol.8, no. 2, pp. 3976-83, 2019.
6. Tan, X., Su, S., Huang, Z., Guo, X., Zuo, Z., Sun, X., & Li, L. "Wireless sensor networks intrusion detection based on SMOTE and the random forest algorithm". Sensors, vol. 19, no. 1, pp. 203, 2019.
7. Awotunde, J. B., Chakraborty, C., & Adeniyi, A. E. "Intrusion detection in industrial internet of things network-based on deep learning model with rule-based feature selection", Wireless communications and mobile computing, pp. 1-17, 2021.
8. Jasim, A. D. "A survey of intrusion detection using deep learning in internet of things", Iraqi Journal For Computer Science and Mathematics, vol.3, no.1,pp. 83-93, 2022.
9. Poornima, I. G. A., & Paramasivan, B. " Anomaly detection in wireless sensor network using machine learning algorithm" ,Computer communications, no. 151, pp. 331-337,2021.
10. Gebremariam, G. G., Panda, J., & Indu, S. " Secure Intrusion Detection System for Hierarchically Distributed Wireless Sensor Networks",In 2021 International Conference on Industrial Electronics, Research and Applications (ICIERA) ,pp. 1-6 IEEE, December, 2021.
11. Yang, Y., Zheng, K., Wu, C., & Yang, Y. "Improving the classification effectiveness of intrusion detection by using improved conditional variational autoencoder and deep neural network", Sensors, vol.19, no. 11, pp. 2528, 2019.
12. Thivakaran, T. K., & Sakthivel, T. "GUARD: an intrusion detection framework for routing protocols in multi-hop wireless networks", Wireless Networks, vol. 25, no. 2, pp. 819_36, 2019.

13. Ifzarne, S., Tabbaa, H., Hafidi, I., & Lamghari, N. “Anomaly detection using machine learning techniques in wireless sensor networks”, In Journal of Physics: Conference Series IOP Publishing Vol. 1743, No. 1, pp. 012021, 2021.
14. Godala, S., & Vaddella, R. P. V. “ A study on intrusion detection system in wireless sensor networks”, International Journal of Communication Networks and Information Security, vol.12, no. 1, pp. 127_41, 2020.
15. Stiawan, D., Idris, M. Y. B., Bamhdi, A. M., & Budiarto, R. “CICIDS-2017 dataset feature analysis with information gain for anomaly detection”. IEEE Access, no. 8, pp. 132911-132921, 2020.