World Journal of Advanced Research and Reviews

eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(Research Article)

Network intrusion detection using hybrid approach

Ashwani Attri, Priyanka Gundeboyena, Vaishnavi Chigurla *, Soumika Moluguri and Nithin Kasoju

Department of CSE (Data Science), ACE Engineering College, Hyderabad, Telangana, India.

World Journal of Advanced Research and Reviews, 2025, 25(02), 507-515

Publication history: Received on 25 December 2024; revised on 31 January 2025; accepted on 02 February 2025

Article DOI: https://doi.org/10.30574/wjarr.2025.25.2.0367

Abstract

This project presents a new approach to network security by combining two types of detection techniques: signature-based and anomaly-based. The signature-based method helps catch known threats by recognizing attack patterns, while the anomaly detection technique, powered by machine learning (specifically Isolation Forest), identifies unusual or new network behaviors that might signal emerging threats. After rigorous testing with benchmark datasets, the system has shown to be more accurate and generates fewer false alarms than traditional methods. It also includes useful features like storing detected anomalies for later review and sending real-time alerts to ensure prompt responses. This research emphasizes how blending these detection methods can make network intrusion systems more effective, with potential future improvements like integrating real-time monitoring or deep learning for even better performance. The findings are currently being prepared for publication.

Keywords: Network intrusion Detection; Machine Learning; Isolation Forest; Signature and Anomaly based detection

1. Introduction

In today's digital world, where cyberattacks are becoming more frequent and sophisticated, ensuring robust network security has never been more critical. Traditional intrusion detection systems (NIDS), while effective against known threats, often struggle to identify new or evolving attack patterns. This gap in security is especially concerning as organizations rely on digital infrastructures for everything from sensitive data storage to real-time communication.

To address these challenges, this project introduces a hybrid NIDS that combines the best of both worlds: signature-based detection and anomaly-based detection. The signature-based method catches familiar attack patterns, while the anomaly detection, powered by machine learning (specifically the Isolation Forest algorithm), uncovers new or unusual network behavior that could indicate an emerging threat.

Through rigorous testing with benchmark datasets, the system has shown improved detection accuracy, a reduction in false positives, and a stronger ability to identify both known and unknown threats. Additionally, features like persistent anomaly storage and real-time alerting make it a practical and effective tool for security professionals. By merging traditional and advanced detection techniques, this project demonstrates how hybrid systems can provide more comprehensive protection in today's fast-evolving cyber landscape.

Looking ahead, the system could be further enhanced with real-time monitoring and deep learning integration, offering even greater adaptability and reliability. The research is currently being prepared for publication, contributing to the ongoing advancements in cybersecurity.

* Corresponding author: Ch. Vaishnavi

2. Related Work

As cyber threats have become more advanced and diverse, traditional intrusion detection systems (IDS) that rely on signature-based methods have shown limitations in detecting new or unknown attacks. To address this, anomaly-based detection techniques have gained popularity, as they are capable of identifying deviations from established network behavior. Among these, the Isolation Forest algorithm has emerged as a particularly effective tool for anomaly detection. Introduced by Liu et al. (2008), the Isolation Forest algorithm works by isolating anomalies through recursive partitioning of data, which makes outliers easier to identify. This method is highly efficient for detecting anomalies in high-dimensional datasets, such as network traffic data, making it ideal for network intrusion detection.

The strength of the Isolation Forest lies in its ability to efficiently identify anomalous data points without requiring a labeled dataset. Unlike traditional anomaly detection techniques, which often rely on distance or density-based methods, Isolation Forest isolates instances of interest by constructing binary trees, making it computationally efficient, even with large datasets. This feature is particularly important in real-time network security applications, where rapid detection and response are critical. Isolation Forest has proven to be effective at identifying malicious behaviors that deviate from normal network patterns, without requiring prior knowledge of attack signatures.

Several studies have demonstrated the effectiveness of Isolation Forest in network intrusion detection. For instance, *Liu et al. (2008)* highlighted its ability to efficiently detect anomalies in large datasets, a key challenge in network traffic analysis. The algorithm has also been applied to identify a wide range of network attacks, such as Denial of Service (DoS) attacks, port scanning, and data exfiltration. In particular, its efficiency in high-dimensional spaces, where other anomaly detection methods struggle, has made it an appealing choice for modern IDS systems.

In terms of performance, Isolation Forest has shown impressive results in comparison to other anomaly detection techniques. For example, *Ahmed et al. (2016)* evaluated several machine learning algorithms for intrusion detection and found that Isolation Forest outperformed traditional methods, both in terms of detection accuracy and speed. Its ability to detect outliers in complex, high-dimensional data without requiring extensive computational resources makes it a powerful tool for scalable, real-time intrusion detection systems.

One of the primary advantages of using Isolation Forest is its ability to work with unsupervised data, which is common in real-world network environments where attack signatures are not always available. The algorithm doesn't need prior training on labeled attack data, allowing it to detect previously unseen or "zero-day" attacks. This makes Isolation Forest particularly useful for detecting emerging threats that signature-based systems may fail to catch. Furthermore, the algorithm's simplicity in implementation and its ability to produce interpretable results contribute to its popularity in practical network security applications.

To evaluate the performance of IDS models using Isolation Forest, researchers often rely on benchmark datasets such as KDD Cup 1999, NSL-KDD, and CICIDS. These datasets contain labeled network traffic data, which allows for the testing of various machine learning algorithms, including Isolation Forest. Studies using these datasets have shown that Isolation Forest is not only efficient but also effective in reducing false positives and accurately detecting a variety of network attacks. *Liu et al. (2017)* demonstrated that by applying Isolation Forest to these benchmark datasets, they were able to achieve superior results compared to other machine learning techniques, particularly in terms of identifying previously unseen attack patterns.

3. Existing System

Traditional intrusion detection systems (IDS) have relied on preset rules or statistical thresholds to identify known attacks or abnormal network activity. Signature-based IDS, for example, works by comparing incoming network traffic to a list of known attack patterns. When it finds a match, it triggers an alert. On the flip side, anomaly-based IDS look for any behaviors that deviate from the established “normal” behavior of the network, using statistical models or machine learning to flag anything unusual.

But both of these methods have their drawbacks. Signature-based systems are excellent at catching familiar, known threats but struggle when faced with new or more sophisticated attacks. Anomaly-based systems, though better at spotting unfamiliar threats, can be too sensitive, often generating a lot of false positives whenever there are small deviations from normal activity—even if they’re harmless. To solve these issues, hybrid systems have been developed that combine both signature-based and anomaly-based techniques, offering a more balanced approach for detecting a wider range of threats with greater accuracy.

Even with these improvements, traditional IDS still face significant challenges. Cyber threats are evolving rapidly, and many systems just can’t keep up. Some still miss complex or subtle attacks, while others may generate too many false alarms, both of which can compromise network security.

4. Proposed Model

The proposed hybrid detection model for the Network Intrusion Detection System (NIDS) integrates signature-based and anomaly-based detection techniques for a multi-layered security approach. The system begins by capturing and preprocessing network traffic, filtering irrelevant data, and extracting features like packet size, protocol, and IP addresses. The signature-based module scans traffic for known attack patterns using a database of predefined signatures to identify threats such as DDoS, malware, or buffer overflow attacks. This method excels at detecting known vulnerabilities but struggles with new or evolving threats.

The anomaly-based module models normal traffic behavior using statistical analysis or machine learning techniques and flags deviations as potential zero-day attacks, insider threats, or sophisticated attacks. Algorithms like k-means or Isolation Forest analyze these deviations. The hybrid decision-making engine correlates alerts from both modules, reducing false positives and negatives. It assigns confidence scores to each alert, prioritizing critical threats. For instance, anomalies detected by both modules with high confidence trigger higher-priority alerts.

The model includes a feedback loop to update the signature database and refine anomaly detection models, ensuring adaptability to evolving threats. By combining the speed of signature-based detection with the flexibility of anomaly-based detection, the hybrid model improves detection accuracy, reduces false positives, and enhances adaptability to new and emerging threats.

5. Methodology

The proposed Network Intrusion Detection System (NIDS) is designed using a modular architecture to ensure efficiency, scalability, and maintainability. Each module performs a specific task, from collecting and preprocessing data to detecting and responding to intrusions.

5.1. Data Collection

This module captures network traffic from various sources, such as packet capture tools and mirrored ports, providing the raw data needed for analysis. It collects key parameters like IP addresses, ports, and protocols.

5.1.1. Dataset Selection:

For effective intrusion detection, the system relies on high-quality datasets. One such dataset is the KDD Cup 1999, a widely recognized benchmark that includes a mix of normal and attack traffic. By carefully preprocessing this data to remove inconsistencies, the system improves its ability to accurately identify cyber threats and adapt to evolving attack patterns.

5.1.2. Data Preprocessing:

The raw data is cleaned and standardized in this module. It handles missing values, removes duplicates, and normalizes the data, ensuring consistency before further processing.

5.2. System Architecture

The NIDS uses machine learning to analyze network traffic, identifying malicious patterns in real time. It collects, processes, and refines data for accuracy before deploying a trained model for continuous monitoring and proactive threat detection

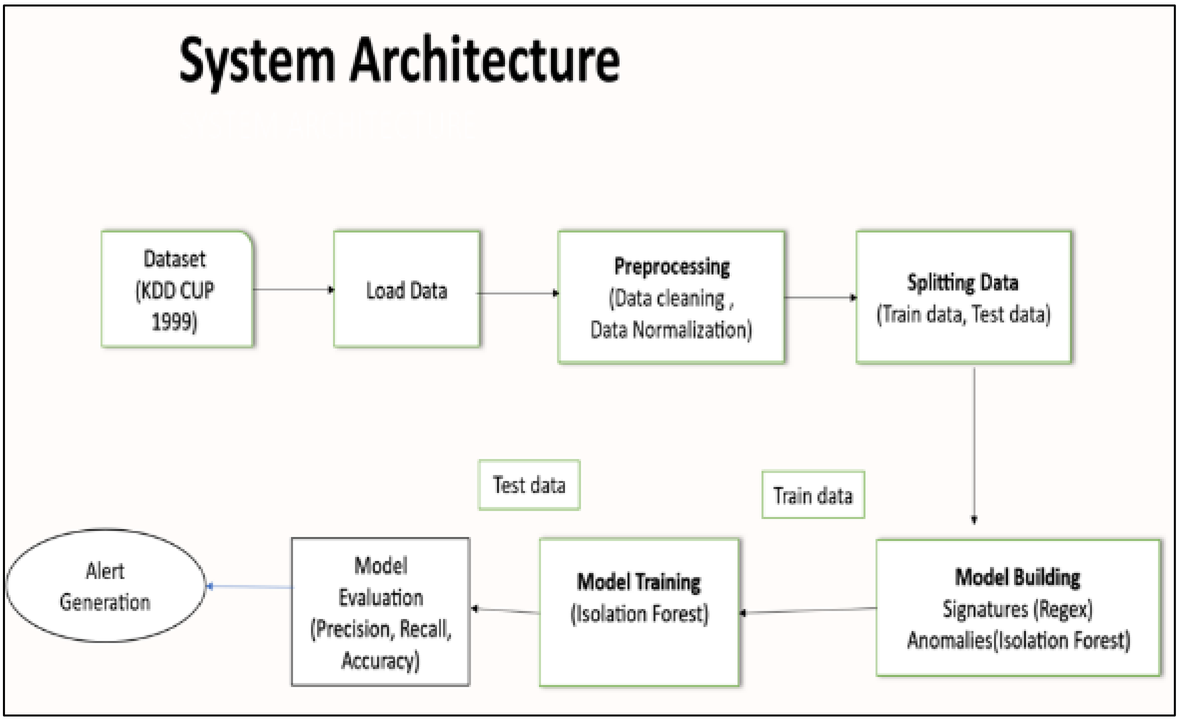

System Architecture
Dataset (KDD CUP 1999)
Load Data
Preprocessing (Data cleaning , Data Normalization)
Splitting Data (Train data, Test data)
Train data
Test data
Model Building Signatures (Regex) Anomalies(Isolation Forest)
Model Training (Isolation Forest)
Model Evaluation (Precision, Recall, Accuracy)
Alert Generation

Figure 1 System Architecture

5.2.1. Feature Engineering Module:

The Feature Engineering Module enhances the NIDS by identifying and extracting significant features from preprocessed data. It employs techniques like PCA for dimensionality reduction and uses feature selection to focus on the most relevant attributes. This optimization improves detection accuracy and model efficiency.

5.2.2. Signature-Based Detection module:

This module captures network traffic from various sources, such as packet capture tools and mirrored ports, providing the raw data needed for analysis. It collects key parameters like IP addresses, ports, and protocols.

5.2.3. Anomaly-Based Detection Module:

Using unsupervised machine learning, this module detects abnormal behavior by creating a baseline of standard network traffic. Any deviation from this baseline is flagged as potential anomalous activity, useful for identifying zero-day attacks or unknown threats.

5.2.4. Hybrid Detection Module:

Combining the strengths of both signature-based and anomaly-based methods, this module enhances detection accuracy by leveraging rule-based techniques for known attacks and machine learning for new or evolving threats.

5.2.5. Evaluation and Performance Metrics Module:

This module evaluates the NIDS's effectiveness using metrics like accuracy, precision, recall, and F1-score. It helps identify strengths and weaknesses in detection methods, ensuring continuous improvement and effective adaptation to new threats.

5.2.6. Alerting Module:

This generates real-time notifications upon detecting intrusions or suspicious activities. Alerts are promptly sent to system administrators via various communication channels such as email, SMS, or a centralized security dashboard. These alerts contain critical information about the detected threat, enabling swift incident response and mitigation.

5.3. Model Development

The model is the core component of the system, responsible for learning and predicting emotions.

5.3.1. **Isolation Forest Algorithm**:

The Isolation Forest algorithm is an unsupervised machine learning method used in the Network Intrusion Detection System (NIDS) to detect anomalies in network traffic. It operates by isolating observations through the creation of decision trees based on random feature splits; anomalies, being less frequent and distinctly different from normal data points, are easier to isolate. The implementation process begins with data collection using tools such as Scapy or pre-captured PCAP files, followed by extracting relevant features, including packet size and connection duration. The Isolation Forest model is then trained on normal traffic data, tuning parameters such as the number of trees and the expected proportion of anomalies. Once trained, the model is applied to real-time network traffic to identify anomalies and generate alerts for any suspicious patterns detected. This method is particularly effective for uncovering potential threats like DoS attacks or unauthorized access attempts, leveraging its adaptive learning capabilities to enhance detection performance in dynamic network environments

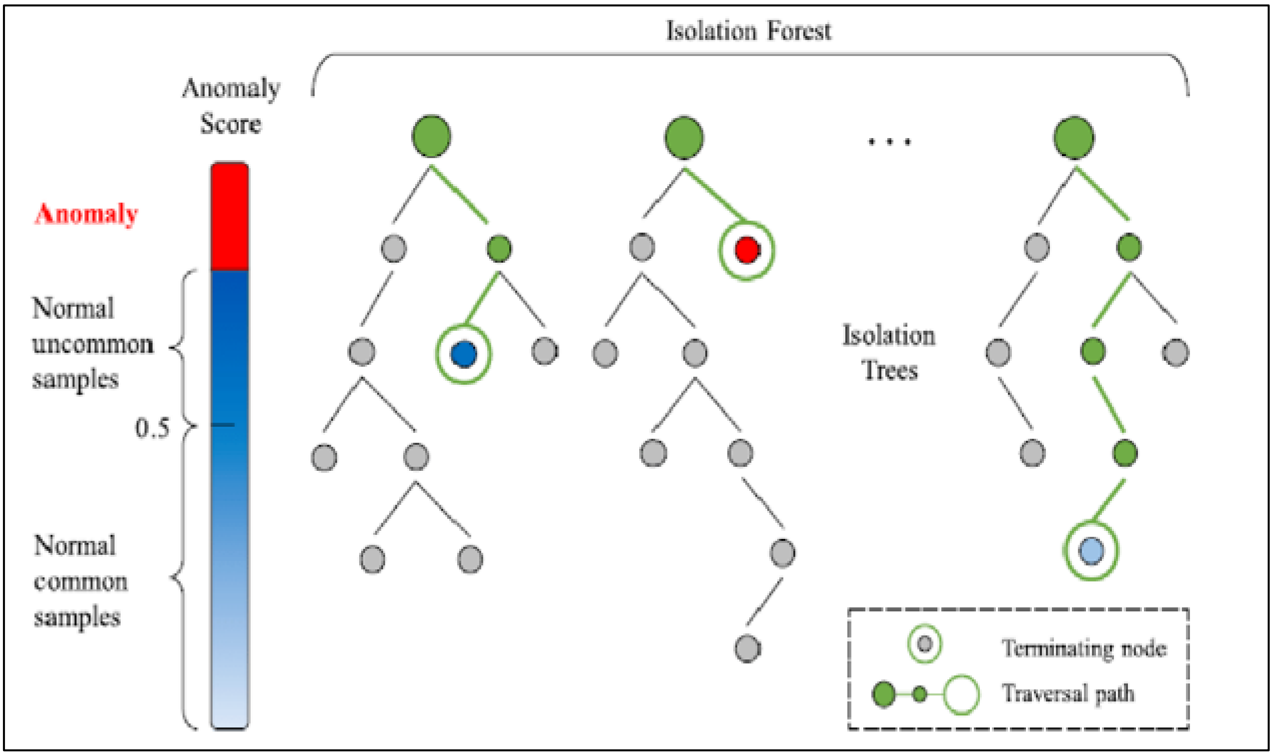


Figure 2 Isolation Forest

5.4. Model Building

5.4.1. Validation

Split the data into training, validation, and test sets. Monitor metrics like accuracy and loss on the validation set to identify and mitigate overfitting. Use techniques like dropout layers to improve generalization.

5.4.2. Training

The training process begins with dataset preparation, where the data is split into 80% training and 20% testing. To ensure consistency in model performance, feature scaling is applied using StandardScaler(), which normalizes the numerical attributes of network traffic. The Isolation Forest algorithm is then trained on normal network traffic data to learn typical patterns and detect deviations effectively. The model is configured with n_estimators=100 to build multiple decision trees and contamination=0.1 to estimate the proportion of anomalies in the dataset.

5.4.3. Testing

Once the model is trained, it is evaluated on the 20% test dataset. The same feature scaling transformation is applied to maintain consistency. The model predicts whether network traffic is normal (1) or anomalous (-1) based on learned patterns. To align predictions with the dataset labels, anomalies detected by the model (-1) are mapped to 1 (indicating an attack), and normal traffic is mapped to 0. The performance of the model is then assessed using accuracy, precision, recall, and F1-score to determine its effectiveness in identifying intrusions.

5.5. Intrusion Detection

5.5.1. Detection Mechanism:

Intrusion detection is all about identifying suspicious or malicious activities within network traffic. To make this process more effective, the system uses a hybrid detection approach that combines the strengths of both signature-based and anomaly-based detection methods. This way, it can catch both known cyber threats and new, evolving attacks.

5.5.2. Signature-Based Detection:

This method works like a security guard checking a list of known threats. It scans network traffic for specific attack patterns, such as SQL injection, DDoS, or malware exploits. If a match is found, an alert is triggered instantly. While this approach is great at detecting well-documented attacks, it may struggle to identify new threats that don't yet have a signature.

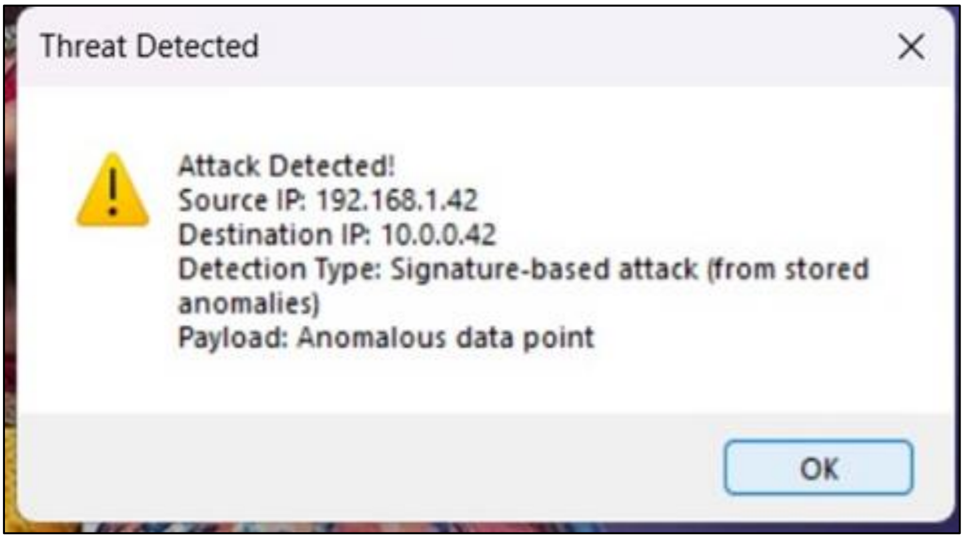
5.5.3. Anomaly-Based Detection:

To catch unknown attacks, the system also relies on machine learning. The model, trained on normal network behavior, continuously monitors traffic and flags anything that doesn't look right. If a data pattern significantly deviates from what's expected, it's treated as a potential threat. This helps detect zero-day attacks, insider threats, or sophisticated cyber intrusions that wouldn't be caught by traditional methods.

5.5.4. Confidence Scoring:

Not every unusual activity is necessarily an attack, so the system assigns a confidence score to each detection. This score helps prioritize the most serious threats while filtering out less certain ones to reduce false alarms. By fine-tuning these thresholds, the system ensures that only the most relevant and high-risk intrusions are flagged for investigation.

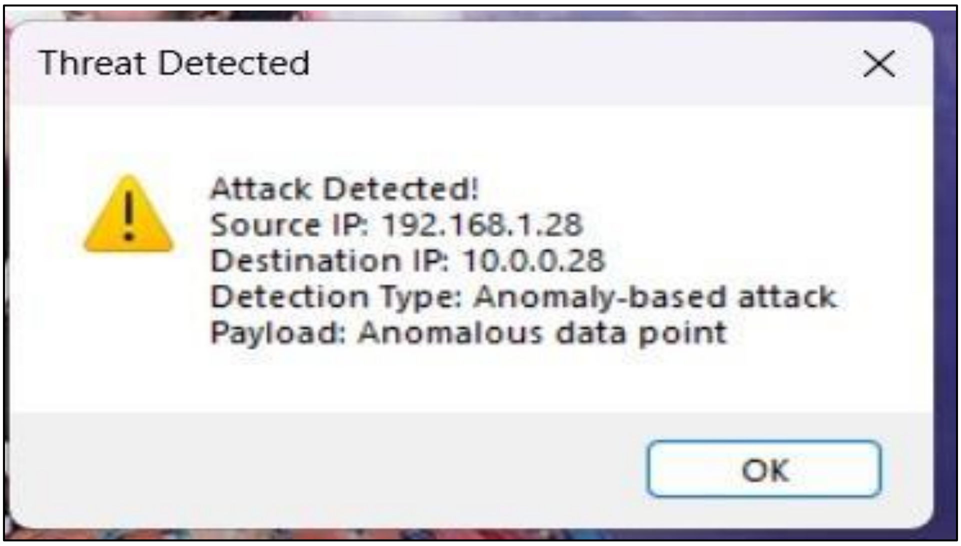
6. Results



Threat Detected

Attack Detected!
Source IP: 192.168.1.42
Destination IP: 10.0.0.42
Detection Type: Signature-based attack (from stored anomalies)
Payload: Anomalous data point

Figure 3 Detecting stored anomalies as signatures



Threat Detected

Attack Detected!
Source IP: 192.168.1.28
Destination IP: 10.0.0.28
Detection Type: Anomaly-based attack
Payload: Anomalous data point

Figure 4 Detecting anomalies

7. Conclusion

This study introduces a hybrid Network Intrusion Detection System (NIDS) that combines signature-based and anomaly-based detection to effectively identify both known and unknown network threats. Tested using the KDD CUP 1999 dataset, the system uses signature-based detection to quickly spot familiar attacks, while the Isolation Forest model helps uncover new and unseen threats. The system provides real-time alerts with important details, such as source and destination IP addresses and attack types, and has the ability to improve over time by learning from past anomalies. Although the results are promising, the system's performance has only been evaluated on the KDD CUP 1999 dataset, leaving its effectiveness in real-world scenarios uncertain. Additionally, exploring unsupervised or semi-supervised learning techniques could further boost its accuracy and adaptability. Overall, this NIDS offers a comprehensive solution for network security, and ongoing research will be key to refining it for future threats and dynamic network environments.

Compliance with ethical standards

Acknowledgments

The author expresses sincere gratitude to Mr. Ashwani Attri and Priyanka Gundeboyena for their valuable contributions to data collection, model evaluation, and insightful discussions that helped shape this research.

Disclosure of conflict of interest

No conflict of interest to be disclosed.

Statement of Ethical approval

This research was conducted in accordance with ethical guidelines to ensure transparency, integrity, and responsible handling of data.

References

[1] KDD Cup 1999 Data. (1999). Retrieved from https://www.kaggle.com/datasets/galaxyh/kdd-cup-1999-data .

[2] A. Anupama and R. R. Prasad, "Hybrid Intrusion Detection System," 2023 International Conference on Quantum Technologies, Communications, Computing, Hardware and Embedded Systems Security (iQ-CCHESS), KOTTAYAM, India, 2023, pp. 1-6, doi: 10.1109/iQ-CCHESS56596.2023.10391328

[3] J. Shi, Y. Lin, Z. Zhang and S. Yu, "A Hybrid Intrusion Detection System Based on Machine Learning under Differential Privacy Protection," 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), Norman, OK, USA, 2021, pp. 1-6, doi: 10.1109/VTC2021-Fall52928.2021.9625540.

[4] R. Zhao, Y. Mu, L. Zou and X. Wen, "A Hybrid Intrusion Detection System Based on Feature Selection and Weighted Stacking Classifier," in IEEE Access, vol. 10, pp. 71414-71426, 2022, doi: 10.1109/ACCESS.2022.3186975.

[5] Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation Forest. 2008 Eighth IEEE International Conference on Data Mining, 413-422. https://doi.org/10.1109/ICDM.2008.17 .

[6] Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. Journal of Network and Computer Applications, 60, 19-31. https://doi.org/10.1016/j.jnca.2015.11.016 .

[7] Liu, F. T., Ting, K. M., & Zhou, Z. H. (2017). Isolation-based anomaly detection. ACM Transactions on Knowledge Discovery from Data (TKDD), 11(1), 1-39. https://doi.org/10.1145/3130511 .

[8] Tavallaee, M., Bagheri, E., Lu, W ., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, 1-6. https://doi.org/10.1109/CISDA.2009.5356528 .

[9] Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2018). Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. International Conference on Information Systems Security and Privacy (ICISSP), 108-116. https://www.scitepress.org/Papers/2018/66398/66398.pdf .

Author’s short biography

Mr. Ashwani Attri: Mr. Ashwani Attri, has Completed his B.Tech and M.Tech in CSE from IIT Kharagpur, He worked in IT Sector as Software Engineer and he is Currently working as Assistant Professor, Department of CSE(Data Science), ACE Engineering College. He aims to inspire students and contribute to advancements in technology through my work.	
Priyanka Gundeboyena : Priyanka Gundeboyena is a final-year B.Tech student at ACE Engineering College, specializing in Computer Science and Engineering (Data Science). My academic journey has been shaped by a strong commitment to exploring data-driven solutions for real-world challenges. Over the course of her studies, she has developed a profound interest in areas such as cybersecurity, machine learning and networking.	
Vaishnavi Chigurla : I am Vaishnavi Chigurla , a final-year B.Tech student at ACE Engineering College, specializing in Computer Science and Engineering (Data Science). My academic journey has led me to explore the dynamic fields of cybersecurity, machine learning, artificial intelligence, and network intrusion detection systems (NIDS). With a keen interest in innovation, I strive to bridge the gap between data science and real-world applications, continuously expanding my knowledge and expertise.	

Soumika Moluguri: Soumika Moluguri is a final-year B.Tech student at ACE Engineering College, specializing in Computer Science and Engineering (Data Science). Her academic journey has led to explore diverse domains, with a particular focus on web development, machine learning, and artificial intelligence. Skilled in building dynamic and efficient web solutions, she committed to continuous learning and innovation to bridge the gap between data science and modern web technologies.	
Nithin kasoju : Nithin kasoju is a final-year B.Tech student at ACE Engineering College, specializing in Computer Science and Engineering (Data Science). His academic journey has been shaped by a strong commitment to exploring data-driven solutions for real-world challenges. Over the course of my studies, He has developed a profound interest in areas such as machine learning, artificial intelligence, and networking.	