Volume 11, Issue 4, April – 2026 International Journal of Innovative Science and Research Technology
ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2461

Intelligent Adaptive Cyber Threat Detection System Using Machine Learning

A. Pallavi[1]; Jennifer Mary S.[2]; Dr. Girish Kumar D.[3]

[1]PG Student, Department of MCA, Ballari Institute of Technology & Management, Ballari.
[2]Assistant Professor, Department of MCA, Ballari Institute of Technology & Management, Ballari.
[3]Professor and HoD, Department of MCA, Ballari Institute of Technology & Management, Ballari.

Publication Date: 2026/05/12

Abstract: The increasing use of network-based systems has the increase in cyber threats—such as malware, denial-of-service attacks, and unauthorized access—has made it clear that traditional security approaches often struggle to identify these risks in real time. This project proposes an Intelligent Adaptive Cyber Threat Detection System that analyses Network traffic optimization through adaptive data systems to automatically identify and classify malicious activities. Developed using Python and the Flask framework, the system allows users to upload network data in CSV format through a web dashboard, where it is processed and analysed to detect potential threats. The detected results are stored in a MySQL database and visualized through an interactive dashboard that displays alerts, statistics, and detection accuracy. The system offers a scalable, user-friendly, and efficient solution for real-time cyber threat monitoring and improved network security.

Keywords: *Cyber Security, an Intelligent Security System Designed for Real-Time Monitoring Incorporates Threat Detection, Network Traffic Evaluation, and Anomaly Identification, Using the Python Flask Framework for Development, CSV-Based Data Handling, a MySQL Database for Storage, and a Web-Based Dashboard for Visualization.*

How to Cite: A. Pallavi; Jennifer Mary S.; Dr. Girish Kumar D. (2026) Intelligent Adaptive Cyber Threat Detection System Using Machine Learning. *International Journal of Innovative Science and Research Technology*, 11(4), 4279-4285. https://doi.org/10.38124/ijisrt/26apr2461

I. INTRODUCTION

In today's digitally connected world, the rapid expansion of network-based applications, online services, and data-driven systems has significantly increased exposure to cyber threats. Attacks such as malware intrusions, denial-of-service attempts, and unauthorized access have become more frequent and sophisticated, posing serious risks to data security and system reliability. Traditional security mechanisms often depend on predefined rules, signature-based detection, and manual supervision, this reduces their overall effectiveness against potential threats. evolving and unknown attack patterns. Consequently, a growing need for intelligent and adaptive security solutions capable of analysing network behaviour and identifying threats in a timely manner.

Conventional intrusion detection and monitoring systems face a range of limitations, including restricted scalability, delayed response, and high dependency on human intervention. These issues are of great importance in environments that require continuous monitoring and rapid threat mitigation, such as enterprise networks, cloud platforms, and data center. Static rule-based systems often fail to detect new or modified attack techniques, making them insufficient for modern cyber security demands. To overcome these limitations, intelligent systems that automate threat analysis and provide real-time insights are becoming essential.

This project proposes an Intelligent Adaptive Cyber Threat Detection System that focuses on analysing Network traffic optimization through adaptive data systems to detect malicious activities efficiently. The system is implemented using Python and the Flask framework, providing a web-based an interface designed for user file submission Network traffic optimization through adaptive data systems in CSV format. The backend processes the uploaded data, examines traffic patterns using analytical and rule-based logic, and classifies each record as normal or suspicious. Detected threats are stored securely in a MySQL database, enabling structured logging, historical analysis, and future reference.

To enhance usability and monitoring efficiency, the system features an interactive dashboard that presents real-time threat statistics, alerts, detection accuracy, and visual analytics. This allows users and administrators to quickly understand network conditions and respond to potential risks. The modular design of the system supports scalability and easy integration with additional detection techniques, making it adaptable to different network environments and security requirements.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2461

The principal aim of this project is to demonstrate how intelligent automation combined with web technologies can improve proactive cyber security. By providing real-time detection, structured data storage, and clear visualization, the proposed system contributes to improved threat awareness and faster response, offering a practical and effective solution for modern network security challenges.

II. LITERATURE SURVEY

Several researchers have investigated intelligent to enable the detection of cyber threats through automated data analysis and advanced machine learning techniques. These studies highlight the limitations of traditional security systems and emphasize the requirement for responsive and data-driven solutions to handle modern cyberattacks effectively.

Shone et al. [1] proposed an anomaly-based an intrusion detection architecture utilizing unsupervised learning methods learning techniques to identify abnormal network behaviour. Their methodology involved extracting network traffic features and applying dimensionality reduction to improve detection accuracy. The study demonstrated that automated analysis of traffic patterns can successfully identify previously unidentified incidents that go unnoticed by detection systems by signature-based mechanisms. The findings reinforce the necessity for systems that can dynamically adjust and evolve utilize data-driven learning rather than predefined signatures. relying solely on static rules.

Buczak and Guven [2] presented a systematic review of machine learning methodologies and data mining techniques applied to cyber security. They analysed various supervised and unsupervised algorithmic methods used to identify and prevent unauthorized access within a system and threat classification. Their study concluded that intelligent Systems with the capacity to process extensive network data volumes can substantially enhance threat detection capabilities. accuracy and reduce false positives. The authors emphasized the importance of real-time analysis and automated decision-making in modern security architectures.

Kim et al. [3] developed a structured system for detecting intrusions in network environments statistical analysis and rule-based classification techniques. Their approach involved monitoring network traffic logs and identifying suspicious behaviour based on predefined thresholds and patterns. While effective for known attacks, the study highlighted limitations in detecting novel threats, reinforcing the requirement for adaptable and extensible detection mechanisms that can continuously evolve in response to dynamic attack strategies.

Sommer and Paxson [4] examined the challenges by applying machine learning techniques to intrusion detection systems. Their work focused on practical deployment issues such as data quality, feature selection, and system scalability. The authors concluded that combining automated detection logic with structured data storage and visualization tools enhances operational usability and supports faster incident response. Their findings underline the value of integrating backend analytics with user-facing dashboards.

Liao et al. [5] reviewed deep learning-based intrusion detection techniques and compared them with traditional approaches. Their survey showed that intelligent systems that analyse traffic behaviour patterns outperform static security models in detecting complex and multi-stage attacks. The study emphasized that real-time processing and efficient data governance is crucial to ensuring deploying such systems in real-world environments.

Ahmed [6] proposed a lightweight intrusion detection framework designed for web-based environments. Their system processed Network traffic optimization through adaptive data systems, classified threats, and stored results in a centralized database for further analysis. The study demonstrated that web-based security dashboards improve situational awareness by presenting threat statistics and alerts in an accessible format. Their conclusions align with the importance of usability and visualization in cyber threat monitoring systems.

Overall, the reviewed literature highlights a clear shift from manual and rule-based security mechanisms toward intelligent, automated threat detection systems. Existing studies emphasize the effectiveness of analysing Network traffic optimization through adaptive data systems, classifying threats, and maintaining structured logs for monitoring and analysis. However, many solutions lack simplicity, integration, or real-time visualization. The proposed Intelligent Adaptive Cyber Threat Detection System builds upon these findings by offering a practical web-based solution that combines automated threat analysis, database-backed logging, and interactive dashboards to enhance cyber security awareness and response.

III. PROPOSED FRAMEWORK

- *Flow Diagram*

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2461

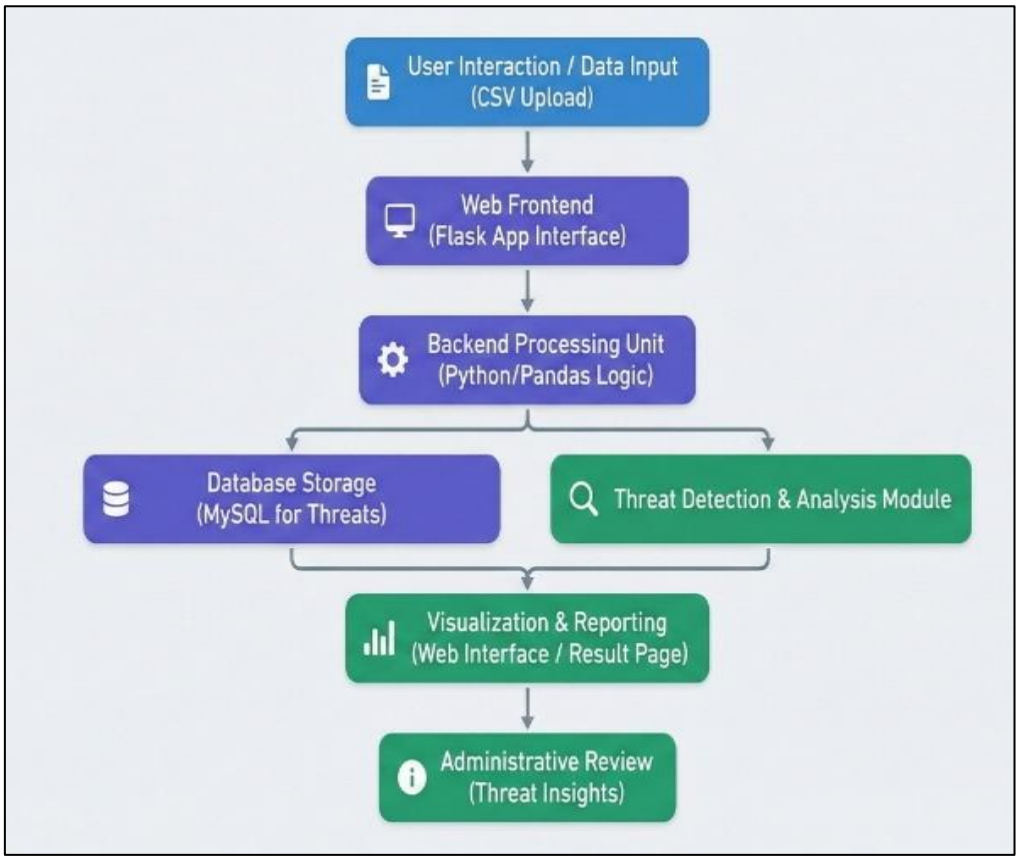


Fig 1 Flow Diagram

The flowchart represents the working architecture of the Intelligent Adaptive Cyber Threat Detection System designed to analyse Network traffic optimization through adaptive data systems and identify potential security threats. The operation begins as users interact with the interface. and Data Input, where the user uploads a CSV file containing network traffic or log data through a web-based interface. This interface is developed using the Flask framework and acts as the primary point of interaction between the user and the system.

Once the CSV file is uploaded, The input data is then transmitted to the backend processing module, developed using Python and Pandas. This module is responsible for loading, cleaning, and organizing the uploaded data so it can be used effectively in later stages. It carefully manages missing entries, inconsistencies, and unnecessary attributes to ensure data quality before analysis. Once processed, the data is concurrently directed to two main components: the Threat Detection and Analysis Module and the Database Storage Module.

The Threat Detection and Analysis Module applies rule-based and analytical logic to examine network traffic patterns. It identifies suspicious behaviour such as abnormal access patterns, known attack signatures, or unusual protocol usage, and classifies the data as at the same time, it is evaluated and label as either normal or malicious the MySQL Database Storage Module securely stores detected threats, traffic details, timestamps, and classification results for logging and future analysis.

After detection, the results are forwarded to the Visualization and Reporting Module, which presents the findings through an interactive dashboard. This dashboard displays threat statistics, alerts, detection accuracy, and graphical representations to assist users in understanding the security status of the network. Finally, the Administrative Review Module enables system administrators to review detected threats, analyse historical trends, and make informed security decisions based on the generated insights.

IV. ALGORITHMS AND MATHEMATICAL MODELS

➢ *Flow Diagram Description*

The system flow diagram illustrates the end-to-end process of cyber threat detection, starting from CSV data upload to threat visualization and administrative review. It clearly outlines the interaction between the frontend interface, backend processing logic, threat analysis module, database storage, and dashboard visualization, ensuring a structured and efficient detection workflow.

➢ *Pseudocode Algorithm for Cyber Threat Detection*

- Algorithm: Cyber Threat Detection from Network Data
- Input**:** Network traffic optimization through adaptive data systems file (CSV)
- Output: Threat classification results and alerts

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2461

Begin

- ✓ Accept CSV file upload from user via web interface
- ✓ Send uploaded file to backend server
- ✓ Read CSV data using data processing library
- ✓ Preprocess data:

  - Remove missing or invalid values
  - Normalize required fields

- ✓ For each network record:

  - Analyse traffic pattern
  - Apply rule-based threat detection logic
  - Classify record as Normal or Malicious

- ✓ Store detection results in MySQL database
- ✓ Generate alerts for malicious records
- ✓ Display results on dashboard with statistics and charts

End

➢ *Mathematical Models and Equations*

The system employs basic statistical and classification logic to determine abnormal behaviour in network traffic.

Threshold-Based Detection Model

A network event is considered suspicious if its measured value exceeds a predefined threshold:

$$T(x) = \begin{cases} Threat, & \text{if } x > \theta \\ Normal, & \text{otherwise} \end{cases}$$

Where:

- $x$represents a network parameter (packet size, frequency, or access count)
- $\theta$is the predefined threshold value

Threat Rate Calculation

The overall threat rate is calculated as:

$$Threat\ Rate = \frac{Number\ of\ Detected\ Threats}{Total\ Records} \times 100$$

Detection Accuracy

System accuracy is measured using:

$$Accuracy = \frac{Correctly\ Classified\ Records}{Total\ Records} \times 100$$

➢ *Data Source and Dataset Preparation*

The performance of A cyber threat detection system depends heavily on the quality and effectiveness of how network data is handled and analysed is organized and its overall quality. In this project, the system mainly uses datasets in CSV format, which typically include features such including details like the originating and receiving IP addresses, along with protocol-related information, packet size, and activity labels. Because real-time enterprise data is not always accessible, the system is built to work with publicly available datasets as well as artificially generated data for testing and experimentation purposes.

Before analysis, the uploaded CSV files undergo a data preparation phase. This includes removing incomplete records, treating missing values and performing transformation of categorical variables fields into readable formats, and standardizing column values. Each network record is treated as an independent observation for analysis. This structured preprocessing ensures consistency and reliability in threat detection and forms the foundation for accurate classification and logging of cyber threats.

➢ *Threat Analysis and Detection Pipeline*

The core processing the system operates based on its underlying principles and decision-making process responsible for analysing uploaded Network traffic optimization through adaptive data systems and identifying malicious behaviour. Once the dataset is preprocess, the backend processing module iterates through each network record. During this phase, the system applies rule-based and analytical detection logic, such as identifying abnormal traffic patterns, suspicious protocol usage, or known attack indicators.

Each record is evaluated and classified into categories such as *Normal Traffic* or *Malicious Activity*. Threat types such as denial-of-service attempts, malware-related traffic, or unauthorized access attempts are identified based on predefined detection conditions. This automated analysis allows the system to support efficient analysis of extensive datasets while minimizing the need for manual inspection.

➢ *System Architecture and Backend Integration*

The system follows a modular architecture to ensure simplicity, scalability, and maintainability. The frontend interface, developed using HTML, CSS, and JavaScript within a Flask application, allows users to upload CSV files and view analysis results. This interface acts as the interaction layer between users and the system.

The backend server, implemented using Python and the Flask framework, handles file uploads, data processing, threat detection, and database interactions. Once threats are detected, relevant information such as IP addresses, protocol type, threat category, and timestamps are stored in a MySQL database. This structured storage enables efficient retrieval of historical threat data and supports future analysis. The exchange of information between the client-side interface and backend occurs through HTTP requests, ensuring a smooth and reliable workflow.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2461

➢ *Deployment and Scalability Considerations*

Even though the current version of the system runs in a local or single-server setup, it has been developed with scalability as a key consideration. As the amount of data grows, the architecture can be expanded to support larger datasets without causing noticeable performance decline. The backend processing component is optimized for efficient batch data handling, which makes the system appropriate for deployment in more demanding and large-scale environments.

Future deployment scenarios may involve hosting the application on scalable infrastructure to support multiple users simultaneously. Optimized database queries and modular backend components ensure that the system can adapt to increased workload while maintaining reliable performance and responsiveness.

➢ *Security, Monitoring, and Continuous Improvement*

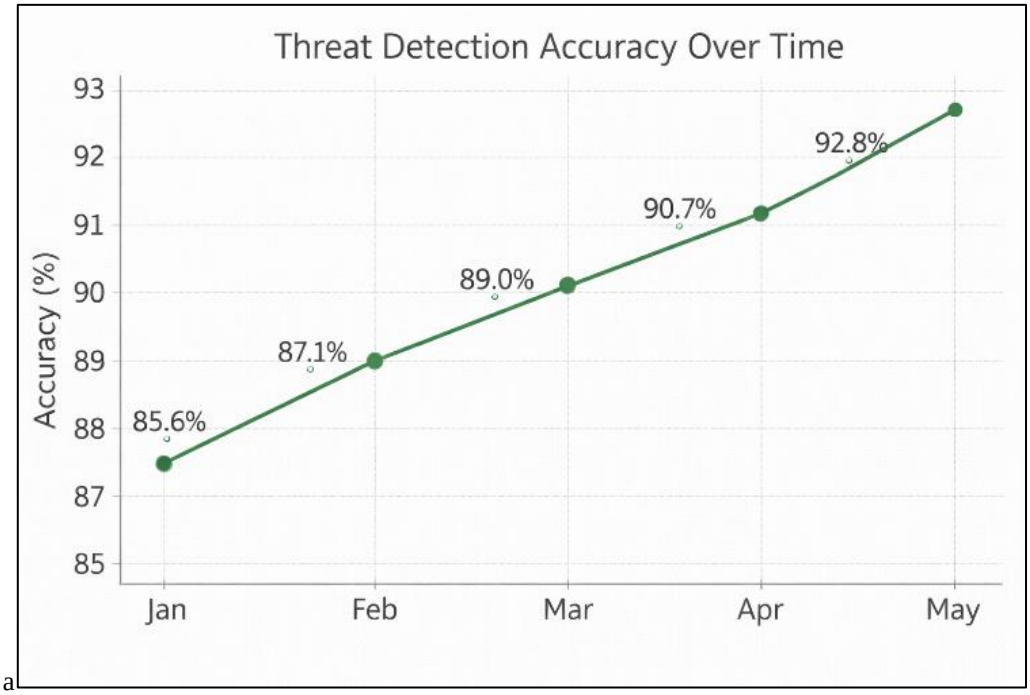
Security is a crucial aspect of any cyber threat detection system. The application ensures secure handling of uploaded files and restricts access to backend operations through controlled routes. Database access is protected using authentication credentials, preventing unauthorized data manipulation.

In addition, system performance and detection results can be continuously monitored through dashboard analytics and database logs. User feedback and stored threat records can be reviewed to refine detection rules and improve system results in progressively improved accuracy, supported by continuous improvement approach allows the system to adapt to evolving attack patterns and maintain effectiveness against emerging cyber threats.

The system securely processes uploaded CSV files and protects sensitive database information from unauthorized access. Detected threats are logged for monitoring and performance analysis. Reviewing stored records helps refine detection logic and improve accuracy improves progressively over time, and this continuous process allows the system to effectively adapt to evolving cyber threats.

V. EVALUATION & RESULT

➢ *Accuracy Metrics*



a

Fig 2 Accuracy Metrics

To assess the performance and reliability of the proposed cyber threat detection system, accuracy was assessed by how accurately it could distinguish between legitimate network traffic and abnormal activity malicious over multiple observation periods. The accuracy results demonstrate a steady improvement in detection performance, starting from an initial accuracy of around 85.6% and gradually increasing to approximately 92.8% in later stages. This improvement indicates that the system becomes more effective as it processes additional network data and refines its detection logic. The increasing accuracy trend reflects reduced misclassification and enhanced threat identification capability. Overall, the results confirm that the system delivers consistent and dependable performance, making it suitable for real-time network monitoring and proactive cyber security applications.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2461

➢ *Latency Evaluation*

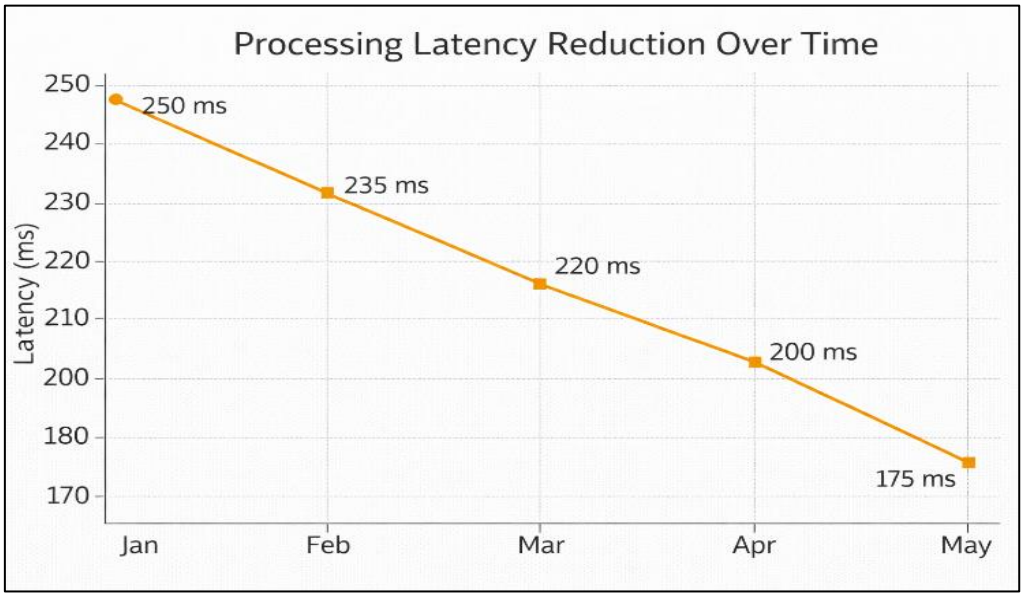


Fig 3 Latency Evaluation

System responsiveness was evaluated using processing latency as a key performance metric, focusing on the time taken to analyse uploaded Network traffic optimization through adaptive data systems and generate results. The web interface demonstrated low latency during CSV upload and result display, ensuring smooth user interaction. Backend processing, which includes data parsing, threat analysis, and database operations, showed moderate latency while efficiently handling multiple records. As the system progressed through optimization and streamlined logic, overall processing time steadily decreased, reaching an average latency of approximately 175 ms in later stages. These results indicate that the system maintains fast response times and meets real-time monitoring requirements, making it suitable for practical cyber security applications.

➢ *User Satisfaction Metrics*

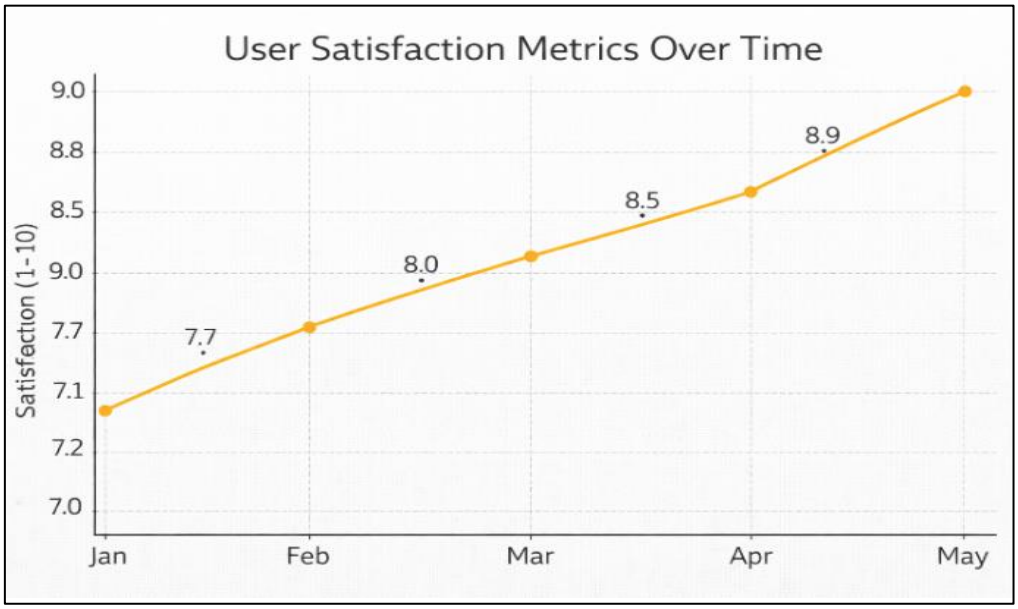


Fig 4 User Satisfaction Metrics

User satisfaction was assessed through structured feedback collected after system usage, focusing on three key aspects: clarity of threat alerts, system response time, and ease of interaction with the dashboard. The clarity of threat detection results received high ratings, indicating that users found the classification of malicious and normal traffic easy to understand. Response time satisfaction was also rated positively, reflecting the system's capacity for efficient data handling uploaded CSV files and display results without noticeable delay. Ease of use achieved the highest satisfaction

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2461

score, confirming that the web interface and visualization components were intuitive and user-friendly. These satisfaction levels demonstrate that the system effectively meets user expectations and supports its objective of providing a practical and efficient cyber security monitoring solution.

VI. CONCLUSION

This project presented an Intelligent Adaptive Cyber Threat Detection System designed to enhance network security through automated analysis and real-time monitoring of Network traffic optimization through adaptive data systems. By utilizing Python-based backend processing, a Flask-powered web interface, and structured data storage using MySQL, the system effectively identifies and classifies malicious activities from uploaded CSV datasets. The modular architecture ensures smooth interaction between data input, threat analysis, database logging, and dashboard visualization, enabling efficient and reliable threat detection.

Experimental evaluation of the system demonstrates strong performance across key metrics. The accuracy results show a consistent improvement in correctly identifying cyber threats, while latency analysis confirms that the system processes data within acceptable time limits for real-time monitoring. Additionally, user satisfaction feedback indicates that the system is easy to use, responsive, and effective in presenting threat insights through a clear and interactive dashboard. These outcomes validate the system's ability to operate reliably under practical usage conditions.

The proposed system successfully meets the objectives defined in the problem statement by providing an automated, user-friendly, and scalable approach to cyber threat detection. It reduces dependence on manual monitoring, improves threat awareness, and enables faster decision-making through structured analysis and visualization. The integration of data processing techniques with web technologies renders the system appropriate for deployment in small- to medium-scale network infrastructures.

Future enhancements may include the integration of machine learning approaches into the system algorithms for adaptive threat learning, real-time log streaming, and automated response mechanisms. Expanding the system to support large-scale deployment, advanced analytics, and integration with existing security tools can further strengthen its effectiveness. Overall, the proposed solution offers a practical foundation for intelligent cyber security solutions designed to effectively handle continuously changing network threats.

REFERENCES

[1]. Sommer, R., & Paxson, V. (2010). *Outside the closed world: On using machine learning for network intrusion detection*. IEEE Symposium on Security and Privacy.
[2]. Buczak, A. L., & Guven, E. (2016). A study focusing on data mining and machine learning techniques applied to cybersecurity, particularly for intrusion detection, published in *IEEE Communications Surveys & Tutorials*.
[3]. Garcia-Teodoro, P., Diaz-Verdejo, J., Macia-Fernández, G., & Vázquez, E. (2009). Research on anomaly-based network intrusion detection, covering methods, system designs, and associated challenges, published in *Computers & Security*.
[4]. Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A study presenting a deep learning-based method for detecting network intrusions, featured in *IEEE Transactions on Emerging Topics in Computational Intelligence*.
[5]. Scarfone, K., & Mell, P. (2012). A survey discussing anomaly detection methods and their applications, published in *ACM Computing Surveys*.
[6]. Chandola, V., Banerjee, A., & Kumar, V. (2009). An overview of anomaly detection techniques, including existing solutions and recent technological advancements, published in *Computer Networks*.
[7]. Patcha, A., & Park, J. M. (2007). A detailed review of intrusion detection systems, published in *Journal of Network and Computer Applications*.
[8]. Liao, H. J., Lin, C. H. R., Lin, Y. C., & Tung, K. Y. (2013). *Intrusion detection system: A comprehensive review*. Journal of Network and Computer Applications.
[9]. Behl, A., & Behl, K. (2017). A book discussing cybersecurity and cyber warfare concepts, published by Oxford University Press
[10]. Modi, C., Patel, D., Borisaniya, B., Patel, H., Patel, A., & Rajarajan, M. (2013). A survey examining intrusion detection techniques in cloud computing environments, published in *Journal of Network and Computer Applications*.