International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

HYBRID AI-BASED FRAMEWORK FOR REAL-TIME CYBER THREAT DETECTION AND PREVENTION USING ISOLATION FOREST AND RULE-BASED INTELLIGENCE

[1]Dandem . Ajay Kumar , [2] Muntha. Raju , [3] Yerram .Sneha

[1] Department of computer science and engineering, Nalla Malla Reddy engineering college, Hyderabad, Telangana,India.
[2] Professor ,Department of computer science and engineering, Nalla Malla Reddy engineering college, Hyderabad, Telangana,India.
[3] Assistant Professor. Department of computer science and engineering, Nalla Malla Reddy engineering college, Hyderabad, Telangana,India.
[1] ajaydandem123@gmail.com ,[2] raj.jntu9@gmail.com , [3] yerramsneha@gmail.com

Abstract

The increasing use of computer networks and online services has resulted in a growing number of cybersecurity threats. Traditional security mechanisms mainly depend on predefined rules and known attack signatures, which may not effectively identify new or unusual patterns of network activity. This paper presents an AI-powered cyber threat detection and prevention dashboard that combines machine learning with rule-based detection for continuous network security monitoring. The proposed system uses the Isolation Forest algorithm to identify abnormal network behavior and a rule-based mechanism to detect activities such as port scanning, repeated login failures, suspicious URL access, and unusually large data transfers. Network events are processed and analyzed to generate alerts and assign risk levels according to the detected threat characteristics. A Streamlit-based interactive dashboard presents network events, threat alerts, risk information, and traffic patterns through tables and graphical visualizations. The system also provides attack simulation capabilities for testing and includes a simulated IP-blocking mechanism to demonstrate preventive response. The integration of anomaly detection, rule-based analysis, real-time visualization, and response support provides a practical framework for improving cybersecurity monitoring and reducing the effort required for manual analysis. The experimental evaluation demonstrates the applicability of the proposed approach for identifying suspicious network activities and supporting timely security decisions. The centralized dashboard further helps administrators observe emerging threat patterns and prioritize security incidents based on their risk levels. The proposed framework can also serve as a foundation for future integration with real network traffic and advanced threat intelligence sources.

Keywords: *Artificial Intelligence, Cybersecurity, Machine Learning, Isolation Forest, Anomaly Detection, Network Threat Detection, Real-Time Monitoring, Rule-Based Detection, Threat Visualization, Cyber Threat Prevention*

I. INTRODUCTION

The rapid growth of digital networks, cloud-based applications, and online services has changed the way organizations store, process, and exchange

information. This increased dependence on connected systems has also created more opportunities for cybercriminals to exploit network and application vulnerabilities. Attacks such as unauthorized access, port scanning, suspicious web activity, repeated login attempts, and abnormal data transfers can cause serious security risks. Therefore, continuous observation of network activity and early identification of suspicious behavior have become important requirements for maintaining a secure computing environment [1], [2].

Conventional cybersecurity solutions generally depend on predefined signatures, fixed rules, and previously identified attack patterns. These techniques are useful for detecting familiar threats, but their effectiveness can decrease when attackers introduce new techniques or modify existing attack strategies. Unknown or unusual activities may not match the patterns stored in conventional security systems. Previous studies have consequently explored machine learning and data-mining techniques as alternative approaches for identifying abnormal network behavior and improving intrusion detection capabilities [3], [4].

Machine learning can examine network-related characteristics and identify patterns that are difficult to recognize through manually defined rules. Anomaly detection is particularly useful in situations where complete labeled datasets containing every possible type of attack are not available. The Isolation Forest algorithm provides an unsupervised approach for identifying observations that differ from the general behavior of a dataset. Its ability to detect unusual observations makes it useful for analyzing network events in cybersecurity applications [5].

However, machine learning-based detection alone may not identify every security condition effectively. Certain threats have recognizable characteristics that can be represented through explicit security rules. For this reason, combining anomaly detection with rule-based analysis can provide a broader detection mechanism. In the proposed system, rule-based checks are used to identify activities such as port scanning, login failures, suspicious URL access, and unusually large file transfers, while the machine learning component examines network behavior for potential anomalies [6], [7].

The increasing quantity of network events also creates difficulties for security administrators. Examining large volumes of raw logs manually can consume considerable time and may delay the identification of important incidents. A visual monitoring interface can make this process easier by presenting network events, alerts, risk levels, and traffic information in an organized manner. Real-time dashboards can provide security personnel with a consolidated view of network conditions and help them focus their attention on events requiring immediate investigation [8], [9].

This research proposes an AI-powered cyber threat detection and prevention dashboard that brings together machine learning-based anomaly detection, rule-based threat identification, real-time monitoring, alert generation, visualization, and prevention support.

The implemented system generates network events for controlled testing and extracts relevant characteristics from these events for analysis using the Isolation Forest algorithm. At the same time, predefined detection rules examine individual events for recognizable suspicious patterns. The outputs from both mechanisms are used to determine threat levels and generate security alerts.

The dashboard is implemented using Python-based technologies, with Streamlit providing the interactive interface and Scikit-Learn supporting the machine learning component. Pandas and NumPy are used for data processing, while Plotly is used to present network activity and anomaly information visually. The system also provides an attack simulation facility for testing different threat scenarios and demonstrates a prevention workflow through simulated IP blocking and response logging. These implementation details are consistent with the uploaded project report.

The main purpose of the proposed framework is to provide a practical security monitoring environment that can identify suspicious network activities while reducing the effort associated with manual analysis. By combining behavioral anomaly detection with explicit security rules, the system provides complementary mechanisms for recognizing different types of threats. The centralized dashboard further enables administrators to observe alerts, examine risk information, and take appropriate preventive actions. The framework can serve as a foundation for future development involving live network traffic, external threat-intelligence sources, cloud environments, and more advanced learning models.

II LITERATURE SURVEY

Artificial intelligence and machine learning have increasingly been investigated for cybersecurity because of their ability to analyze large quantities of network information and identify patterns associated with suspicious activity. Conventional security mechanisms generally depend on previously defined signatures and rules, which can be effective for recognized attacks but may have limitations when the behavior of an attack changes. Research in intelligent intrusion detection has therefore focused on developing methods that can recognize abnormal network behavior and support faster security analysis [1].

Buczak and Guven [1] reviewed several data-mining and machine-learning techniques used for cyber intrusion detection. Their study discussed the application of different learning methods for identifying malicious network behavior and highlighted important challenges associated with data preparation, feature selection, algorithm selection, and system evaluation. Their findings indicate that machine learning can provide useful support for security monitoring when appropriate network characteristics are selected for analysis.

García-Teodoro et al. [2] investigated anomaly-based intrusion detection techniques and described how deviations from expected network behavior can be

used to identify potentially malicious activities. Anomaly-based detection is particularly useful when an attack does not have a previously stored signature. However, the effectiveness of such systems depends on establishing appropriate representations of normal behavior and controlling false alarms. These observations are relevant to the development of systems that combine behavioral analysis with additional detection mechanisms.

Liu, Ting, and Zhou [3] introduced the Isolation Forest algorithm as an efficient technique for identifying anomalous observations. The method isolates individual observations through recursive random partitioning and gives greater attention to observations that can be separated more easily. Its unsupervised nature makes it appropriate for situations where obtaining complete labeled attack data is difficult. In cybersecurity applications, this characteristic can be useful for identifying unusual network events without requiring examples of every possible attack.

Modi et al. [4] examined intrusion detection techniques in cloud computing environments. Their work discussed the security difficulties associated with dynamic and distributed infrastructures and reviewed different approaches for identifying malicious activities. The study emphasizes that security mechanisms must be capable of adapting to changing environments rather than depending entirely on fixed attack descriptions. This requirement is also relevant to modern network monitoring systems that may need to process continuously changing traffic patterns.

Javaid et al. [5] investigated the application of deep learning to network intrusion detection. Their work demonstrated the potential of learning-based techniques for recognizing patterns within network traffic. Deep learning approaches can process complex relationships in data, although they may require substantial computational resources and suitable training datasets. This research represents the broader movement toward intelligent models for strengthening network intrusion detection.

Nguyen and Armitage [6] studied machine-learning methods for Internet traffic classification. Their work demonstrated that network traffic characteristics can be analyzed computationally to distinguish different communication patterns. Such research provides a basis for extracting meaningful features from network events before applying machine-learning techniques. In the proposed system, network properties such as protocol, port information, event type, and data-transfer volume are similarly converted into features for analysis.

Kim and Kim [7] investigated a deep-learning-based intrusion detection approach for improving network security. Their study reflects the growing use of intelligent models to identify malicious network behavior and reduce dependence on manually specified detection patterns. Although the proposed project uses Isolation Forest rather than a deep-learning architecture, the study supports the use of automated learning methods for cybersecurity monitoring.

Muthusamy [8] examined the use of artificial intelligence for threat detection in cybersecurity environments. The study emphasizes the role of intelligent techniques in improving the identification of cyber threats and supporting security response. AI-based analysis can assist security personnel by examining network information systematically and highlighting activities that require further investigation.

Kumar et al. [9] investigated AI-driven approaches for cyber threat detection and mitigation. Their work indicates that intelligent security frameworks can assist in recognizing malicious activities and improving the efficiency of cybersecurity operations. The use of AI can reduce the dependence on continuous manual examination of security information and provide additional analytical capabilities for threat monitoring.

Research has also considered real-time monitoring as an important part of intelligent cybersecurity systems. Ravi and Umesh [10] examined machine-learning approaches for real-time network intrusion detection, emphasizing the importance of processing network information promptly. Real-time analysis can reduce the delay between the occurrence of suspicious activity and its identification, which is important when security incidents require immediate attention.

Ahmed et al. [11] discussed AI-based cyber defense mechanisms for modern networks. Their work reflects the increasing requirement for security systems that do more than simply identify threats. Intelligent security platforms are expected to provide useful information for decision-making and assist in responding to detected incidents. This concept is consistent with systems that combine detection, alert generation, visualization, and prevention functions.

Li, Zhang, and Chen [12] investigated real-time intrusion detection using stream analytics and artificial intelligence. Their work highlights the importance of processing continuously generated network information and identifying suspicious patterns with limited delay. Stream-oriented analysis is particularly relevant to monitoring systems in which network events arrive continuously rather than as a single static dataset.

Patel and Shah [13] studied dashboard-based cybersecurity monitoring systems using artificial intelligence. Their research highlights the value of presenting security information through visual interfaces. A dashboard can consolidate alerts, network statistics, and threat information into a common environment, helping administrators understand the current security condition without examining every raw event individually.

Singh and Gupta [14] examined cloud-based AI security frameworks for intrusion detection and prevention. Their work indicates that integrating intelligent detection with prevention capabilities can provide a more comprehensive security architecture. Such frameworks can support the transition from passive monitoring toward more proactive security management.

From the reviewed literature, it can be observed that machine learning provides useful capabilities for detecting network anomalies, while rule-based techniques remain valuable for recognizing clearly defined threat conditions. However, using either approach independently may leave certain detection requirements unaddressed. A combined mechanism can therefore provide complementary capabilities by using behavioral analysis for unusual activities and explicit rules for recognizable threat patterns.

The proposed research builds upon these observations by integrating an Isolation Forest-based anomaly detector with rule-based threat identification. The system analyzes network events, generates alerts, assigns risk levels, and presents the results through an interactive dashboard. It also includes simulated attack generation and IP-blocking functionality to demonstrate a complete detection-to-response workflow. The implementation described in the project report uses Python, Scikit-Learn, Pandas, NumPy, Plotly, and Streamlit to develop this integrated monitoring environment.

III EXISTING SYSTEM

Existing cybersecurity systems commonly use firewalls, Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS), antivirus software, and Security Information and Event Management (SIEM) tools to monitor network activities and identify security incidents. Most of these solutions depend on predefined signatures and security rules to recognize known attack patterns. Although such mechanisms are effective against previously identified threats, they may not recognize new or modified attacks that do not match the stored patterns. As cyberattacks continue to evolve, this dependence on predefined knowledge can reduce the ability of conventional systems to identify previously unseen or unusual network behavior.

Another major limitation is the large volume of network logs and security alerts generated in modern environments. Manual examination of these events can be time-consuming and may increase the possibility of delayed responses or human errors. Conventional systems may also produce false alerts and provide limited behavioral analysis and visualization capabilities. In addition, monitoring, detection, and response functions are often distributed across different security tools, making it difficult to obtain a unified view of network security. These limitations create the need for an intelligent and centralized solution capable of combining behavioral anomaly detection, rule-based analysis, real-time visualization, alert generation, and preventive actions.

IV PROBLEM STATEMENT

The rapid expansion of network-based applications has made it increasingly difficult to monitor security events and identify suspicious activities in a timely manner. Conventional security mechanisms generally depend on predefined signatures and fixed detection rules, which may recognize known threats but can struggle with unusual or newly emerging attack behavior. In addition, the continuous generation of

network logs makes manual examination difficult and can increase the possibility of overlooking important security events.

To address these challenges, there is a need for a centralized and intelligent monitoring system that can examine network events from different perspectives and highlight potentially harmful activities. The proposed work focuses on combining machine-learning-based anomaly detection with predefined security checks to identify both unusual behavior and recognizable threats. The system also aims to provide real-time alerts, risk information, graphical monitoring, and prevention support through a single dashboard, helping administrators analyze security events more efficiently and respond to potential threats in a timely manner.

V PROPOSED SYSTEM

The proposed system is an AI-powered cyber threat detection and prevention dashboard designed to monitor network activities and identify potentially harmful events. It combines machine-learning-based anomaly detection with predefined security rules so that both unusual network behavior and recognizable threat patterns can be identified. The system analyzes information such as source and destination IP addresses, port numbers, communication protocols, transferred data volume, URLs, and event types. These attributes are converted into suitable numerical features before being analyzed by the detection model.

The Isolation Forest algorithm is used as the primary machine-learning technique for detecting abnormal network behavior. The model is initially trained using a baseline set of generated network events and then analyzes recent events to calculate anomaly scores. In addition to the machine-learning component, rule-based checks are used to identify conditions such as port scanning, login failures, suspicious URL access, and unusually large data transfers. The outputs from both mechanisms are combined to determine the risk associated with an event and to generate an appropriate security alert.

The system provides an interactive Streamlit dashboard through which administrators can monitor network events and detected threats in real time. The interface displays recent network activity, anomaly scores, alert information, risk levels, and traffic trends using tables and graphical visualizations. It also provides controls for starting or pausing the event stream, clearing logs and alerts, changing the event generation rate, retraining the anomaly model, and exporting alerts. A simulated IP-blocking facility is included to demonstrate how preventive action can be initiated after a suspicious source is identified.

The proposed framework is evaluated using controlled network-event scenarios containing normal and suspicious activities. Test cases are used to verify event generation, detection of scanning activity, login failures, suspicious URLs, large data transfers, and machine-learning-based anomalies.

VI METHODOLOGY

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

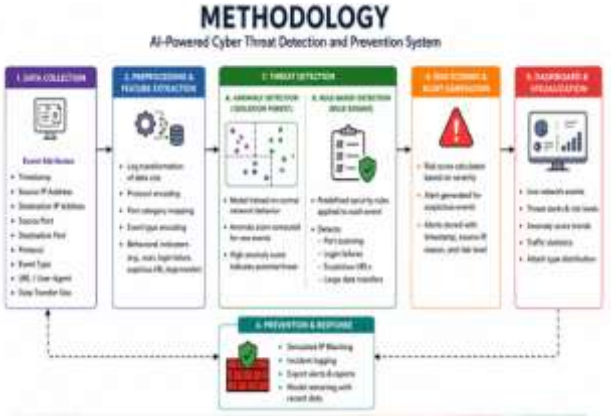

Fig1:AI-Driven Network Threat Detection and Prevention Framework

The proposed methodology follows a sequential AI-powered cybersecurity framework that starts with data collection and proceeds through preprocessing, threat detection, risk assessment, visualization, and response. Initially, network events are collected along with essential attributes such as timestamp, source IP address, destination IP address, source port, destination port, protocol, event type, URL/User-Agent, and data transfer size. The collected network data is then passed to the preprocessing and feature extraction stage, where log transformation is applied to the data size, protocols are encoded, port categories are mapped, and event types are converted into suitable numerical representations. Additional behavioral indicators are extracted, including scan indicators, login failure indicators, suspicious URL indicators, and large data transfer indicators. The processed data is subsequently analyzed by the threat detection module, which consists of two complementary approaches: Anomaly Detection using Isolation Forest and Rule-Based Detection using a Rule Engine. The Isolation Forest model learns normal network behavior and calculates an anomaly score for each network event, enabling the identification of highly abnormal events as potential threats. In parallel, the rule engine applies predefined security rules to detect known suspicious activities such as port scanning, repeated login failures, suspicious URLs, and large data transfers. The results from both detection mechanisms provide the required information for determining whether a network event represents normal or potentially malicious behavior.

The detected events are then forwarded to the risk scoring and alert generation stage, where the results from machine-learning-based anomaly detection and rule-based detection are combined to calculate an overall risk score. Based on the observed security conditions and calculated score, suspicious events are assessed according to their threat severity and appropriate alerts are generated. Each alert is stored with important details such as the timestamp, source IP address, detection reason, and risk level, allowing security administrators to track and investigate incidents. The generated information is presented through the dashboard and visualization module, which provides a centralized view of live network events, threat alerts and risk levels, anomaly-score trends, traffic statistics, and attack-type distribution. Finally, the methodology incorporates a prevention and response mechanism to support further security actions. Simulated IP blocking can be performed for identified malicious sources, while incident logging maintains a record of detected security events. The

system also supports exporting alerts and reports for further analysis and documentation, and model retraining using recent network data enables the detection model to adapt to changing network behavior. Thus, the complete flow from data collection through detection, risk assessment, visualization, and prevention provides an integrated framework for automated cyber-threat detection and response.

VII IMPLEMENTATION

the proposed AI-powered Cyber Threat Detection and Prevention Dashboard was implemented using Python as the primary programming language. Stream lit was used to develop the interactive web-based dashboard for monitoring network activities, displaying alerts, and presenting security-related visualizations. The machine-learning component was developed using the Scikit-learn library, with the Isolation Forest algorithm used for detecting unusual network behavior. Pandas and NumPy were used for data processing, feature preparation, and numerical operations. Plotly Express was used to generate interactive charts and graphs for representing network traffic and detected threats. The Faker library was used to generate synthetic network events for testing and real-time simulation. The system was executed in a Python development environment with the required libraries installed and configured before running the dashboard.

The proposed AI-powered Cyber Threat Detection and Prevention Dashboard was implemented using Python, with Streamlit used to develop the interactive monitoring interface. The system generates simulated network events containing information such as source and destination IP addresses, protocols, destination ports, transferred bytes, event types, user-agent details, and URLs. Different types of activities, including normal traffic, failed login attempts, port scanning, file downloads, and suspicious URLs, are generated to provide varied network conditions for testing. The generated events are then converted into suitable numerical features through preprocessing, where parameters such as transferred bytes, protocols, ports, and event types are transformed into a format that can be processed by the machine-learning model.

For intelligent threat detection, the system uses the Isolation Forest algorithm provided through Scikit-learn to identify unusual network behavior. The model is initially trained using generated network events and evaluates incoming events based on their extracted features. In parallel, a rule-based detection mechanism checks predefined conditions associated with suspicious URLs, large data transfers, port scanning, and failed login attempts. The outputs from the machine-learning model and rule-based mechanism are combined to determine the risk associated with each event. When suspicious activity is identified, an alert is generated containing important information such as the timestamp, source IP, destination IP, event type, transferred data, risk level, and reason for detection.

The detected activities are presented through the Streamlit dashboard using tables, alerts, and interactive visualizations for easier monitoring. The dashboard displays recent network events, detected

anomalies, traffic information, and high-risk alerts, while also providing controls for simulation and model retraining. To reduce repeated notifications, an alert deduplication mechanism is applied to similar events occurring within a short period. The system also provides a simulated prevention mechanism in which suspicious source IP addresses can be blocked and response activities can be recorded. Thus, the implementation integrates network-event generation, feature extraction, AI-based anomaly detection, rule-based threat identification, risk assessment, real-time visualization, alert management, and simulated threat prevention within a single security monitoring platform.

VIII. RESULTS AND DISCUSSION

The developed AI-powered Cyber Threat Detection and Prevention Dashboard was tested using different types of simulated network activities to evaluate the working of the proposed system. The testing mainly focused on network event generation, feature extraction, anomaly detection, rule-based threat identification, risk assessment, alert generation, dashboard visualization, and prevention actions. The system was able to process the generated network events and identify activities that differed from normal traffic. During testing, normal traffic was monitored without unnecessary alerts, while suspicious activities such as port scanning, failed login attempts, suspicious URLs, and large data transfers were identified successfully. The Isolation Forest model was also able to identify abnormal network behavior and contribute to the risk assessment process.

S. No.	Test Case	Expected Result	Obtained Result	Status
1	Application Launch	Application should start successfully	Application launched successfully	Pass
2	Network Event Generation	Network events should be generated	Events generated successfully	Pass
3	Normal Traffic	Normal traffic should be processed without unnecessary alerts	Normal traffic processed	Pass
4	Port Scanning	Port scanning activity should be detected	Port scan detected	Pass
5	Failed Login	Login failure should be identified	Failed login detected	Pass
6	Suspicious URL	Suspicious URL should be identified	Suspicious URL detected	Pass
7	Large Data Transfer	Large transfer should be detected	Large transfer detected	Pass
8	ML Anomaly Detection	Abnormal behavior should be identified	Anomaly detected	Pass
9	Alert Display	Detected threats should be displayed	Alert displayed successfully	Pass

Table 1. Functional Testing Results

The results indicate that the major functional components of the system worked as expected during testing. The network-event generator continuously produced different types of activities, which were then converted into features for further processing. The rule-based mechanism was useful for detecting predefined threat patterns, whereas the Isolation Forest model provided an additional mechanism for identifying unusual behavior. Combining these two approaches allowed the system to consider both known suspicious activities and deviations from normal network behavior.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

S. No.	Network Activity	Detection Method	Result
1	Normal Traffic	Event Processing	Monitored
2	Port Scanning	Rule-Based Detection	Detected
3	Failed Login	Rule-Based Detection	Detected
4	Suspicious URL	Rule-Based Detection	Detected
5	Large Data Transfer	Rule-Based Detection	Detected
6	Unusual Network Behavior	Isolation Forest	Detected

Table 2. Threat Detection Results

The system also performed risk assessment for the detected events. Different threat conditions were assigned different risk levels based on their nature and severity. Suspicious URLs, large data transfers, port scanning, and failed login activities were evaluated by the rule engine, while the machine-learning model contributed to the identification of abnormal events. When a suspicious event satisfied the alert conditions, the system generated an alert containing details such as the timestamp, source IP address, destination IP address, event type, transferred bytes, risk value, and reason for detection.

S. No.	Detected Activity	Risk Category	Alert Status	System Response
1	Normal Traffic	Low	Not Generated	Continued Monitoring
2	Failed Login	Medium	Generated	Security Alert
3	Port Scanning	High	Generated	Security Alert
4	Suspicious URL	High	Generated	Security Alert
5	Large Data Transfer	High	Generated	Security Alert
6	ML-Based Anomaly	High	Generated	Security Alert

Table 3. Risk Assessment and Alert Generation

The Streamlit dashboard provided a convenient interface for observing the network activities and detected threats. Recent network events, traffic information, anomalies, and security alerts were displayed through the dashboard. The visualization component helped present the information in a more understandable form, allowing the administrator to observe changes in network activity and identify high-risk events. The dashboard also included controls for simulation, model retraining, and other monitoring operations.

S. No.	System Component	Observation	Status
1	Network Monitoring	Recent network events displayed	Successful
2	Anomaly Monitoring	Detected anomalies displayed	Successful
3	Security Alerts	Recent alerts displayed	Successful
4	Risk Information	High-risk activities identified	Successful
5	Visualization	Network activity represented graphically	Successful
6	Model Retraining	Retraining option available	Successful
7	IP Blocking	Suspicious IP blocking simulated	Successful
8	Alert Management	Repeated alerts controlled	Successful
9	Data Export	Alert information available for export	Successful

Table 4. Dashboard and Response Results

The prevention component further extended the system from simple threat detection to basic threat response. When suspicious activity was identified, the system supported simulated blocking of the associated source IP address. Alert deduplication was also implemented to reduce repeated alerts for similar

activities occurring within a short period. These features make the dashboard more useful from an administrator's point of view because the detected threats can be reviewed along with the response taken by the system.

The developed prototype provides a useful foundation for further development of an intelligent cybersecurity monitoring system. Future evaluation can be extended by using real network traffic, larger datasets, additional threat categories, deep-learning models, threat-intelligence feeds, and automated incident-response mechanisms. Such improvements can help assess the system under more realistic conditions and provide quantitative performance measurements for comparing different detection techniques.

IX. CONCLUSION & FUTURE SCOPE

CONCLUSION

The developed AI-powered Cyber Threat Detection and Prevention Dashboard provides an integrated approach for monitoring network activities, identifying suspicious behavior, assessing security risks, and generating alerts. The proposed system combines rule-based threat detection with the Isolation Forest machine-learning model, allowing it to identify both predefined suspicious activities and unusual network behavior. During functional testing, the system successfully processed normal and suspicious network events, including port scanning, failed login attempts, suspicious URLs, and large data transfers.

The experimental results show that all 14 tested functional areas successfully passed the evaluation. The system performed network-event generation, feature extraction, anomaly detection, risk assessment, alert generation, dashboard visualization, simulated IP blocking, and alert deduplication as expected. The Streamlit-based dashboard provided a simple interface for observing recent network activities, detected anomalies, risk information, and security alerts. The combination of machine-learning-based anomaly detection and rule-based detection provides a practical foundation for developing an intelligent cybersecurity monitoring solution.

FUTURE SCOPE

The proposed system can be further enhanced by integrating real-time network traffic collected from live network environments instead of relying mainly on simulated events. Larger and more diverse cybersecurity datasets can be used to train and evaluate the anomaly detection model under different network conditions. Future versions can also incorporate additional machine-learning and deep-learning techniques to improve the identification of complex and previously unseen cyber threats.

The system can be extended by integrating external threat-intelligence feeds to obtain updated information about malicious IP addresses, domains, URLs, and known attack patterns. Automated incident-response mechanisms can also be introduced so that high-risk threats can trigger appropriate preventive actions without requiring continuous manual intervention.

More advanced features such as real-time packet analysis, user and entity behavior analysis, improved alert prioritization, and automated investigation can further strengthen the platform.

Future research should also focus on conducting extensive experiments using standard benchmark datasets and real network environments. This would allow the calculation of accuracy, precision, recall, F1-score, false-positive rate, detection time, and other relevant performance measures. Such quantitative evaluation would provide a stronger basis for comparing the proposed approach with existing cybersecurity detection techniques and for developing a scalable, reliable, and intelligent cyber threat detection and prevention platform.

REFERENCES

[1] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in *Proc. 8th IEEE Int. Conf. Data Mining (ICDM)*, Pisa, Italy, 2008, pp. 413–422, doi: 10.1109/ICDM.2008.17.

[2] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation-based anomaly detection," *ACM Trans. Knowledge Discovery from Data*, vol. 6, no. 1, pp. 1–39, 2012, doi: 10.1145/2133360.2133363.

[3] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *Computers & Security*, vol. 28, nos. 1–2, pp. 18–28, 2009, doi: 10.1016/j.cose.2008.08.003.

[4] A. Patcha and J.-M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," *Computer Networks*, vol. 51, no. 12, pp. 3448–3470, 2007, doi: 10.1016/j.comnet.2007.02.001.

[5] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 20, 2019, doi: 10.1186/s42400-019-0038-7.

[6] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *Proc. IEEE Symp. Security and Privacy*, Oakland, CA, USA, 2010, pp. 305–316.

[7] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Trans. Emerging Topics in Computational Intelligence*, vol. 2, no. 1, pp. 41–50, 2018, doi: 10.1109/TETCI.2017.2772792.

[8] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.

[9] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Military Communications and Information Systems Conf. (MilCIS)*, Canberra, ACT, Australia, 2015, pp. 1–6, doi: 10.1109/MilCIS.2015.7348942.

[10] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Information Systems Security and Privacy (ICISSP)*, Funchal, Portugal, 2018, pp. 108–116.

[11] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Computational Intelligence for Security and Defense Applications*, Ottawa, ON, Canada, 2009, pp. 1–6.

[12] W. Tounsi and H. Rais, "A survey on technical threat intelligence in the age of sophisticated cyber attacks," *Computers & Security*, vol. 72, pp. 212–233, 2018, doi: 10.1016/j.cose.2017.11.001.

[13] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, "Deep learning approach for network intrusion detection in software defined networking," in *Proc. Int. Conf. Wireless Networks and Mobile Communications (WINCOM)*, Fez, Morocco, 2016.

[14] S. Revathi and A. Malathi, "A detailed analysis on NSL-KDD dataset for intrusion detection system," in *Proc. IEEE Int. Conf. Recent Trends in Information Technology (ICRTIT)*, Chennai, India, 2013, pp. 897–902.

[15] MITRE, "MITRE ATT&CK: Adversarial tactics, techniques, and procedures," MITRE, 2026. [Online]. Available: MITRE ATT&CK.