International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 10, Issue 04, ISSN No. 2455-2143, Pages 46-49
Published Online August 2025 in IJEAST (http://www.ijeast.com)

A REVIEW ON ANOMALY IDENTIFICATION IN CYBER LOGS THROUGH MACHINE LEARNING TECHNIQUES

Kuldeep Prajapti
Research Scholar
NRI Institute of Research & Technology Bhopal, (M.P.), India

Dr.Devendra Baipai
Associate Professor
NRI Institute of Research & Technology Bhopal, (M.P.), India

Abstract: **With the increasing complexity and volume of cyber attacks, traditional rule-based systems are often inadequate in detecting novel threats. Machine Learning (ML) techniques have emerged as a potent solution for identifying anomalies in cyber security logs, offering the ability to learn from data and detect suspicious patterns with minimal human intervention. This**
This review paper presents a comprehensive survey of existing ML approaches for anomaly detection in cyber security logs. We break down well-known datasets, estimate performance criteria, bandy important difficulties and undiscovered investigation directions, and classify styles into supervised, unsupervised, and semi-supervised orders.

I. INTRODUCTION

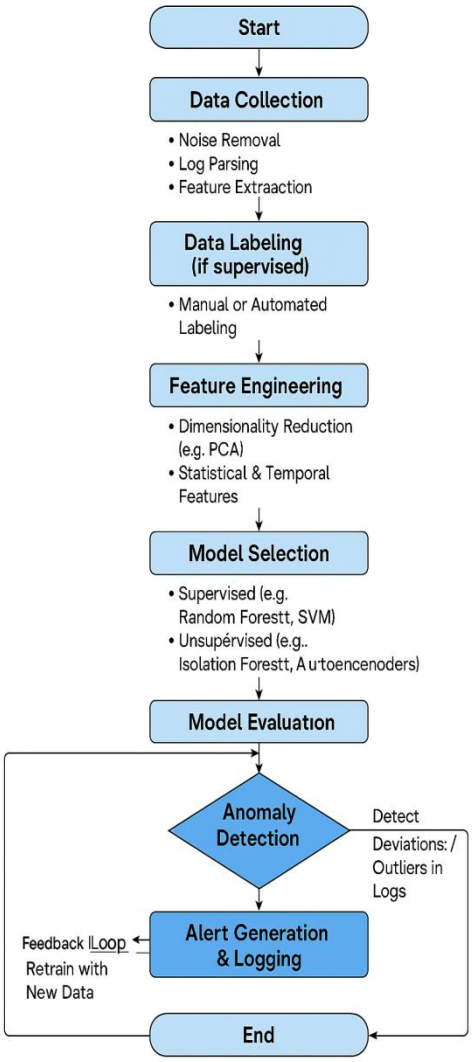
Cyber security has become a critical concern across industries, driven by the growing reliance on digital infrastructure. Log data generated by firewalls, intrusion detection systems, authentication servers, and network traffic analyzers is a valuable resource for identifying security breaches. However, manually sifting through these logs is inefficient and error-prone.
Anomaly detection offers a way to identify deviations from normal behavior that may signify malicious activity. Machine Learning techniques enable automated, scalable, and adaptive detection mechanisms, making them ideal for analyzing large-scale log data

1. Types of Anomalies in Cyber Logs

- Point **Anomalies:** Single log entries that deviate from the norm (e.g., login from an unknown IP).
- Contextual **Anomalies:** Events that are anomalous in a specific context (e.g., access during unusual hours).
- Collective **Anomalies:** A group of related entries that together form an anomaly (e.g., a sequence of failed login attempts).

2. Suggested Procedure

- The flow chart below explains the proposed methodology.



International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 10, Issue 04, ISSN No. 2455-2143, Pages 46-49
Published Online August 2025 in IJEAST (http://www.ijeast.com)

1. Preprocessing

Data Collection: Logs are gathered from various sources including SIEM systems, firewall logs, system authentication logs, and IDS/IPS tools.

Log Parsing: Raw unstructured logs are converted into structured formats (CSV, JSON) for analysis.

Timestamp Normalization: Ensures temporal consistency by aligning timestamps across different time zones and formats.

Noise Reduction: Redundant or irrelevant entries are removed to reduce data clutter.

Log Aggregation: Related events are grouped (e.g., by session or user) to provide a holistic view of activities.

2. Feature Engineering

- Transform raw log entries into meaningful features for ML algorithms.
- Techniques:
    - Statistical features: counts, frequencies, and durations.
    - Categorical encoding: IP addresses, event types, usernames.
    - Time of day and day of week are examples of temporal attributes.
    - Textual features: using TF-IDF or word embeddings for log messages.

3. Anomaly Detection Model

The ML model will be selected based on the nature of available data (labeled or unlabeled):

- Supervised Models (if labeled data is available):
    - Random Forest, XG Boost, SVM, Deep Neural Networks.
- Unsupervised Models (for unlabeled data):
- Auto encoders, K-Means, DBSCAN, One-Class SVM, and Isolation Forest.
- Semi-Supervised Models:
- Only trained on typical data: Variation Auto encoders (VAEs), Deep SVDD, and One-Class SVM.

4. Model Training and Validation

- Train-Test Split or Cross-Validation for supervised models.
- Anomaly Scoring for unsupervised models to rank suspicious events.
- Evaluation Metrics:
    - Accuracy, Precision, Recall, F1-Score.
    - Area under the Curve (AUC).
    - Both True Positive Rate (TPR) and False Positive Rate (FPR) are important.

5. Post-Detection Analysis

- Correlation with threat intelligence feeds.
- Prioritizing alerts according to their severity and historical background
- Explainable AI (XAI) tools like LIME or SHAP to interpret model predictions.

6. Challenges and Future Directions

Despite significant progress, several challenges persist:

- Data Labeling: High-quality labeled data is scarce and often domain-specific.
- Concept Drift: Evolving attack patterns require models to adapt over time.
- Scalability: Handling real-time detection across large networks remains computationally intensive.
- Interpretability: Black-box ML models lack transparency, making them difficult to trust in critical systems.

II. FUTURE RESEARCH

- Integrating federated learning to preserve privacy across distributed networks.
- Developing domain-specific feature extractors.
- Leveraging graph-based models for relational log analysis.
- Developing hybrid models for robust detection that integrate supervised and unsupervised learning.

III. CONCLUSION

Machine Learning has transformed the landscape of anomaly detection in cyber security by enabling intelligent, scalable, and adaptable solutions. By leveraging structured feature engineering, advanced detection models, and post-

analysis interpretability, ML-based frameworks provide a compelling alternative to traditional signature-based methods. However, continued research is needed to address challenges like data quality, interpretability, and real-time performance, paving the way for more resilient and intelligent cyber security systems.

IV. REFERENCES

1. V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM Comput. Surv., vol. 41, no. 3, pp. 1–58, Jul. 2009, doi: 10.1145/1541880.1541882.

2. M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," J. Netw. Comput. Appl., vol. 60, pp. 19–31, Jan. 2016, doi: 10.1016/j.jnca.2015.11.016.

3. G. Kim, S. Lee, and S. Kim, "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection," Expert Syst. Appl., vol. 41, no. 4, pp. 1690–1700, Mar. 2014, doi: 10.1016/j.eswa.2013.08.066.

4. A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," IEEE Commun. Surv. Tutor., vol. 18, no. 2, pp. 1153–1176, 2016, doi: 10.1109/COMST.2015.2494502.

5. A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in Proc. 9th EAI Int. Conf. Bio-inspired Inf. Commun. Technol., 2016, doi: 10.4108/eai.3-12-2015.2262516.

6. M. He et al., "Loghub: A large collection of system log datasets towards automated log analytics," arXiv preprint arXiv:2008.06448, 2020.

7. NSL-KDD Dataset. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html

8. LogPai LogHub Repository. [Online]. Available: https://github.com/logpai/loghub

9. F. Pedregosa et al., "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res., vol. 12, pp. 2825–2830, Oct. 2011.

10. M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you?: Explaining the predictions of any classifier," in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., 2016, pp. 1135–1144, doi: 10.1145/2939672.2939778.

11. S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," arXiv preprint arXiv:1705.07874, 2017.

12. C. Chio and D. Freeman, Machine Learning and Security: Protecting Systems with Data and Algorithms, 1st ed., O'Reilly Media, 2018.

13. B. Xu, D. Zhu, Q. Liu, and H. Xiong, "Detecting anomalies in logs using unsupervised LSTM models," arXiv preprint arXiv:1809.03604, 2018.

14. R. Bertero et al., "Experience report: Log mining using natural language processing and application to anomaly detection," in Proc. IEEE Int. Symp. Software Reliability Eng., 2017, pp. 351–360.

15. W. Xu, L. Huang, and M. J. Nixon, "Detecting large-scale system problems by mining console logs," in Proc. SOSP, 2009, pp. 117–132.

16. H. Duan, T. Li, D. Ge, and T. Zhu, "Anomaly detection using attention-based bidirectional LSTM," in Proc. IEEE INFOCOM WKSHPS, 2020.

17. H. Nguyen and J. Armitage, "Anomaly detection in log data using deep learning," in Proc. Int. Conf. Comput. Sci. Inf. Syst., 2018, pp. 1–6.

18. P. Alipour and A. A. Sadeghzadeh, "System log analysis using deep learning techniques," Computers & Security, vol. 97, 2020, doi: 10.1016/j.cose.2020.101968.

19. K. Brown and J. Richards, "Evaluating One-Class SVM for anomaly detection in cyber logs," in Proc. Int. Conf. Security Softw. Eng., 2021.

20. R. Bansal et al., "A comparative analysis of machine learning algorithms for anomaly detection in security logs," in Proc. ICCC, 2022, pp. 102–108.

21. M. Landauer et al., "Deep learning for anomaly detection in log data: A survey," Mach. Learn. Appl., vol. 12, Jun. 2023.

22. G.-F. Chen et al., "Enhancing log anomaly detection through knowledge graph integration," in Proc. IEEE Int. Conf. Semantic Comput., Feb. 2024, pp. 204–207.

23. X. Liang, L. Li, and H. Peng, "Unsupervised microservice log anomaly detection method based on graph neural network," in Proc. Int. Conf. Swarm Intell., Aug. 2024, pp. 197–208.

24. Z. Li et al., "Graph neural networks based log anomaly detection and explanation," in Proc. IEEE/ACM Int. Conf. Software Eng., May 2024.