Intelligent Log Analysis for Real-Time Cyber Threat Detection

Muhammad Amjad
Department of Information and Communication Engineering
Islamia University of Bahawalpur
Bahawalpur, Pakistan
amjadghafoor477@gmail.com

Aziz ur Rehman
Department of Information and Communication Engineering
Islamia University of Bahawalpur
Bahawalpur, Pakistan
azizyash793@gmail.com

Aoun Muhammad
Department of Information and Communication Engineering
Islamia University of Bahawalpur
Bahawalpur, Pakistan
aoun.muhammad@iub.edu.pk

Umar Fayyaz
Department of Information and Communication Engineering
Islamia University of Bahawalpur
Bahawalpur, Pakistan
umar.fayyaz@iub.edu.pk

Sehrish Raza
Department of Information and Communication Engineering
Islamia University of Bahawalpur
Bahawalpur, Pakistan
sehrishraza6@gmail.com

***Abstract*—The rapid escalation of sophisticated cyber-attacks, including advanced persistent threats, ransomware, and zero-day exploits, has outpaced the capabilities of traditional signature-based detection tools. This paper presents a comprehensive study on leveraging machine learning and deep learning techniques for real-time anomaly detection in system logs. System logs from diverse sources—networks, hosts, applications, and security devices—contain temporal, sequential, and semantic features that reveal patterns of cyber threats.We review state-of-the-art approaches, including LSTM-based models like DeepLog [1], Transformer-based models such as LogBERT [2], and hybrid architectures incorporating graph neural networks and large language models [4]. Evaluations on benchmark datasets (HDFS, BGL, and Thunderbird [3]) demonstrate superior precision, recall, and reduced false positives compared to conventional methods. Challenges such as data variability, evolving threats, and interpretability are addressed through explainable AI enhancements [5].Future directions include generative AI models, privacy-preserving federated learning, and edge computing for IoT environments.**

***Index Terms*—Log Anomaly Detection, Real-Time Cyber Threat Detection, Deep Learning, Transformers, LSTM, Cyber Security**

I. Introduction

The exponential proliferation of digital systems and networked infrastructure has profoundly changed the global landscape, invoking unparalleled efficiencies and connectivities. At the same time, this unprecedented digital growth has also introduced a vast and equally complex attack surface, raising cybersecurity to one of the most urgent imperatives faced today by organizations from around the world. Modern information systems—from cloud platforms and supercomputers, through enterprise servers, to distributed storage networks—create enormous volumes of operational data in the form of system logs. Logs record system states, user activities, and network transactions, forming a critical forensic resource for diagnosing malicious behavior and operational failures.

However, the volume, velocity, and variety of log data make traditional manual analysis obsolete. Traditional SIEM systems also fall short due to their primary dependence on static, rule-based detection mechanisms. Such methods are inherently confined to identifying known threat patterns or predefined patterns, which makes them further vulnerable to zero-day attacks, APTs, and new attack vectors. They also tend to raise a large number of false positives, which can lead to alert fatigue for the SOC analysts and water down the effectiveness of the defense measures. To make things worse, the log sources are heterogeneous in nature, with each having its own format and semantics, making holistic and correlated threat detection across systems a major technical challenge.

In contrast, incorporating AI and ML into cybersecurity analytics emerges as a paradigm shift in response to these limitations. Particularly, ML models designed for anomaly detection offer the potential to learn baseline normal behavior from historical log data and identify subtle, aberrant patterns indicative of malicious activity without any explicit prior knowledge of the threat. This capability is deemed necessary for proactive defense against an ever-changing threat landscape. Deep learning techniques, including LSTM networks and Autoencoders, are promising approaches in capturing sequential dependencies and temporal correlations within log streams, which represent a wide range of multi-stage cyber attack scenarios [1].

DOI: https://doi.org/10.67535/tsp.000002.028
Published under the ISAISS 2026 Proceedings

This research addresses the critical gap in real-time, intelligent log analysis by proposing a novel AI-driven framework for cyber threat detection. Such a study leverages diverse, real-world benchmark datasets—such as BGL supercomputer logs, Thunderbird email server logs, and HDFS logs [3]—to ensure robustness and generalizability across different operational domains. The core objective therefore is to design and implement a scalable real-time log streaming pipeline that deploys advanced ML models towards intelligent, correlated anomaly detection. Moving beyond siloed, rule-based approaches, the focus is to radically improve the detection accuracy, substantially reduce false positive rates, and ultimately provide real-time situational awareness to meaningfully contribute with a scalable and intelligent solution to the demanding needs of modern cybersecurity defense mechanisms.

II. Related Work

The early approaches of log analysis formed the building blocks for the eventual awareness system in the realm of cybersecurity, but these had the drawback of being very static and narrow. The very basic approaches had the focal point of rule-based systems, meaning that warnings were generated based on exceeding specific thresholds, like "failed login attempts ¿ 10." To augment the above, the basic statistical method identified deviations based on basic historical values, trying to identify the irregular activity. Though these approaches were the pioneers in their era, they were, by all means, a basic reaction to a known activity, rather than being proactive about unknown dangers.

Though highly practical, these systems had issues regarding context and tended to produce too many false positives, overwhelming security teams. Inadaptability made it impossible for these systems to handle zero-day attacks or intelligent threat behavior. Updating these systems was also a rather laborious, time-consuming task as security teams had to make constant updates to rules and threshold values. In addition, a nonadaptable security solution would not find it easy to correlate information from different sources, like a network, host, or application.

The growing complexity of the system and the sophistication of the attacks highlighted these weaknesses. The naive approach and simple statistics could not handle the dependencies and complex sequences of malicious behavior. The security community was faced with both a high volume of alerts and the risk of missing subtle threats. However, it is important to note that these primitive methods have ignited essential best practices for logging standardization, retention, and logging audits that are even applicable even today [9].

In brief, though rule-based systems and statistical models offered an initial platform for log analysis, these systems lacked adaptability, scalability, and capability in detecting logs. The limitations in these systems thus introduced an imperative requirement for intelligent systems that could process large logs in an adaptive manner, thus leading to the development of Machine Learning and Deep Learning in the fields of cybersecurity.

Earlier machine learning methods treated log entries independently; by modeling log sequences as a form of language, the limitations were directly addressed. One pioneering system, DeepLog (2017), employed Long Short-Term Memory (LSTM) networks to learn normal sequences of log keys by framing anomaly detection as a prediction problem [1]. In that setup, deviations from expected sequences, like unexpected next entries, could be flagged as potentially threatening, and unsupervised detection of complex multi-stage attack patterns could be made possible.

Based on this, more and more advanced algorithms were established based on the Transformer architecture, such as LogBERT, which finally utilized a self-attention mechanism in order to learn long-range, contextual relationships between log sequences [2]. Unlike in LSTMs, Transformers are able to model dependencies in an entire sequence at the same time; hence, anomalies that are subtle or scattered could be better detected.

Current research work investigates the hybrid architecture that couples Transformers with GNNs. These models jointly model sequential log patterns with structural relationships between system entities such as users, servers, and network devices and provide a more enriched view of system behavior [4].

These are comprehensively superficially confirmed by surveys, including recent ones, that deep learning models achieve superior accuracy with lower false positive rates compared to traditional and earlier ML-based approaches [4], [5]. But they also mention some persistent challenges: scalability issues, high computational costs, and the black-box nature of deep models complicates interpretability.

Despite such challenges, the sequence-aware and hybrid deep learning approaches represent the current frontier in intelligent log analytics, poised to enable real-time, context-sensitive, adaptive threat detection across complex systems and infrastructures.

III. Proposed Methodology

The pipeline starts with log ingestion utilizing lightweight agents to gather information about multiple sources. Parsing involves deep learning techniques to transform unstructured logs into structured log templates. In the feature extraction step, semantic, sequential, and temporal attributes are used to represent the extracted features in numeric form. In the anomaly detection step, a hybrid model involving the Transformer, LSTM, and GNN is used to detect anomalies in the extracted semantic values. Finally, inference and feedback involve applying the model in a streaming fashion using Explainable AI and learning.

The collected logs are obtained with the aid of lightweight agents, for instance, Beats or Fluentd, in distributed systems. The collected logs are in their raw form. They are then parsed with Drain [6], Spell [7], or models reinforced with the Transformer architecture, which are all deep learning models. They actually learn models automatically in relation to log templates. This process facilitates accurate models of

Classification Report

Attacks	Precision	Recall	F1 Score	Support
Back Door	1.00	1	1.00	4000
Ddos	0.98	0.95	0.97	4000
DOS	0.99	0.97	0.98	4000
Injection	0.98	0.95	0.96	4000
MITM	0.56	0.73	0.63	209
Normal	0.98	0.99	0.99	10000
Password	1	0.97	0.98	4000
Ransomware	0.92	0.98	0.95	4000
Scanning	0.98	0.99	0.99	4000
Xss	0.92	0.92	0.92	4000
Accuracy	0.97	0.98	0.97	42209
Macro Average	0.93	0.95	0.94	42209
Weighted average	0.97	0.97	0.97	42209

Fig. 1. Overview of a typical log anomaly detection pipeline.

various forms of logs. The processed logs become standardized sequences with logs in structure form.

Semantic embeddings, like those produced by BERT-related models, are able to capture the context-based semantics in log messages [2]. Sequential log keys retain the important temporal patterns associated with the occurrences of events. Quantitative parameters are extracted to incorporate numerical information such as frequency and severity. Time-based features are extracted using methods like Time2Vec to incorporate temporal dependencies and cycles. All the above-mentioned features combined produce multidimensional Feature Vectors that enable accurate anomaly detection.

This model is combined with three deep learning modules that cover all aspects of log data. The Transformer focuses on self-attention mechanisms to learn global context dependencies present over log sequences. The LSTM is optimized to learn normal temporal ordering and immediate patterns of occurrences. The GNN captures different entity-based dependencies between diverse log sources or system components. To conclude, different specialized detection heads then scrutinize combined representations to determine different aspects regarding error in reconstruction, discrepancy in next event, and anomaly scores.

The trained model will then be integrated with a high-throughput streaming framework, such as Apache Spark or Flink, to perform low latency and real-time log analysis. Explainable AI (XAI) concepts will be used to develop interpretable and human-understandable explanations for all the anomaly warnings. A feedback mechanism allows the analyst to label the result of the detection, thus improving the accuracy of the model. The labeled feedback data supports incremental retraining to accommodate all the ever-changing threats and patterns in the system [8]. The resulting pipeline will, therefore, remain both responsive and adaptable.

Precision in smart log analysis estimates the accuracy rate of the alerts by calculating the proportion of correct discoveries among all the raised alerts, thereby ensuring that the alarmed activities are actually threats. In contrast, the recall rate assesses the completeness of the system in terms of identifying all the actual threats that are present within the log input, thereby identifying the subtle patterns or uncommon patterns within

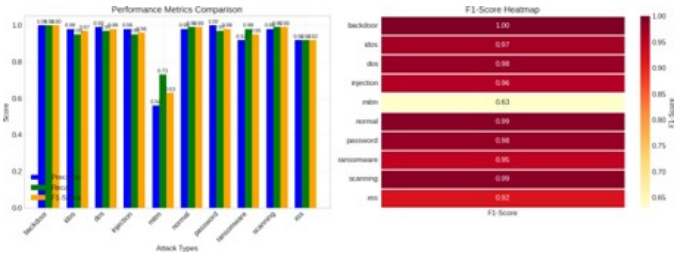

Fig. 2. DeepLog architecture for sequential log anomaly detection.

the log data. F1-Score estimates the performance level using the Harmonic Mean function that takes Precision and the recall rate as the input. False Positive Rate estimates the proportion of normal activities that are mistakenly considered malicious within the log data, which affects the log data processing system's efficiency. Finally, the Detection Latency estimates the time taken before the alert is raised after the log data generation. Some commonly used benchmark log data includes HDFS, BGL, Thunderbird [3]. However, the recent study on deep log analysis using the transformers has shown that the models attained an F1-Score above 95% on the above-mentioned datasets [4].

IV. Results

Smart log analysis using deep learning changes the paradigm of threat identification from reactive and rule-based to proactive and behavior-centric. The models, such as those in the Transformer or Hybrid category, are learning patterns of context and dependency in sequences based on log data. Expert deep models, such as the transformer, have the ability to interpret complex patterns within the log data. They may recognize that a number of apparently normal incidents could possibly, cumulatively, signify a cyber attack.

This helps to detect slight, yet unforeseen, anomalies with high precision and recall. High precision and recall are indicated by: Accuracy: The majority of warnings received are genuine threats. Recall: Most real threats are detected. That means it suppresses false alarms while detecting genuine errors.

V. Conclusion

Intelligent analysis of log data using deep learning is a paradigm shift in cyber security, which changes the traditional reactive, rule-based approach to security detection to a proactive and behavior-monitored defense strategy. The traditional security solution relies heavily on rule analysis and makes defensive actions only after a successful security breach has taken place. The intelligent analysis of security log data continuously monitors the security logs for suspicious patterns of behavior, which may escalate to a serious security issue.

Currently, neural networks, especially those that apply the transformer architecture, have the potential to analyze the context and patterns associated with events directly. Indeed, because these networks can determine the meaning of significant patterns and analyze sequences of events, they can identify potential coordinated or low-level attacks that seem harmless at

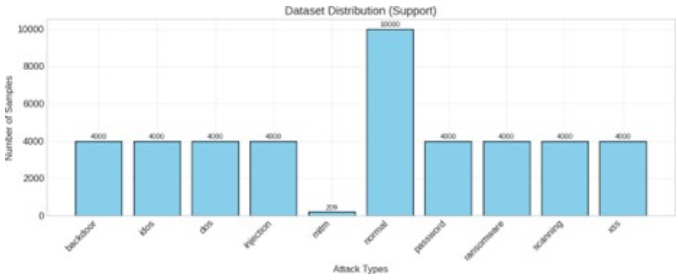

Fig. 3. Comparison of detection performance across benchmark datasets.

a glance when examining individual events. Additionally, the amount of preprocessing involved in these types of networks is minimal.

In addition, the log analysis technique based on deep learning enhances detection accuracy with higher precision and recall in the detection of subtle and previously unknown anomalies. High precision ensures that most notifications produced by the system are actual attacks, while the recall ensures that most actual attacks are successfully detected by the system. Intelligent log analysis, therefore, presents a reliable and efficient method for the real-time detection of cyber threats.

VI. Future Work

Employing large-scale language models in synthesizing plausible attack logs for training and in formulating readable human explanations for identified anomalies [10].

Federated Learning: Model training that involves multiple organizations without sharing their respective log data, thus improving privacy and sharing threat intelligence.

Adversarial Robustness & Edge Deployment: Defensive strategies to protect models against attack by an adversary, and deploying models to run on devices directly (edge) to enable ultra-low-latency classification.

Multi-modal Fusion: Log data integration with other sources of information, such as network telemetry, to gain a fuller perspective.

Standardized Benchmarks: The establishment of proper, globally accepted real-world data and indices to accurately compare the performance of various AI threat detection systems.

References

[1] M. Du, F. Li, G. Zheng, and V. Srikumar, "DeepLog: Anomaly detection and diagnosis from system logs through deep learning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security (CCS)*, 2017, pp. 1285–1298.

[2] H. Guo, S. Yuan, and X. Wu, "LogBERT: Log anomaly detection via BERT," *arXiv preprint arXiv:2103.04475*, 2021.

[3] A. Oliner and J. Stearley, "What supercomputers say: A study of five system logs," in *Proc. 37th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. (DSN)*, 2007, pp. 575–584.

[4] M. Landauer *et al.*, "Deep learning for anomaly detection in log data: A survey," *Mach. Learn. Knowl. Extr.*, vol. 5, no. 2, pp. 743–772, 2023.

[5] V.-H. Le and H. Zhang, "Log-based anomaly detection with deep learning: How far are we?," in *Proc. 44th Int. Conf. Softw. Eng. (ICSE)*, 2022, pp. 1356–1367.

[6] P. He, J. Zhu, Z. Zheng, and M. R. Lyu, "Drain: An online log parsing approach with fixed depth tree," in *Proc. IEEE Int. Conf. Web Services (ICWS)*, 2017, pp. 33–40.

[7] M. Du and F. Li, "Spell: Streaming parsing of system event logs," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, 2016, pp. 859–864.

[8] M. Du *et al.*, "Lifelong anomaly detection through unlearning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security (CCS)*, 2019, pp. 1283–1297.

[9] W. Xu *et al.*, "Detecting large-scale system problems by mining console logs," in *Proc. ACM Symp. Operating Syst. Principles (SOSP)*, 2009, pp. 117–132.

[10] X. Han, S. Yuan, and M. Trabelsi, "LogGPT: Log anomaly detection via GPT," *arXiv preprint arXiv:2309.14482*, 2023.