JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Rule-First and Evidence-Driven MLOps Framework for Cybersecurity Anomaly Detection and Investigation

An Integrated Approach to Security Detection, Evidence Analysis and ML Lifecycle Governance

[1]Aayeda Khan, [2]Trupti Asolkar
[1]M.Sc. Computer Science, [2]Asst. Professor
Department of Advanced Computing
[1]Nagindas Khandwala College,
University of Mumbai, Maharashtra, India

Abstract: Modern cybersecurity setups generate amounts of mixed logs and events which makes it really hard to find the threats that are important and look into suspicious activity fast. Rule based detection is still useful because the logic is clear and easy for analysts to believe in while machine learning based anomaly detection can find behavior that no rule was written for. But in reality these two methods usually work in systems so the signals from anomalies rarely connect properly to security evidence, investigation steps or the daily management of the models that create them. This paper introduces AegisML, a rule-first evidence-driven MLOps framework made for cybersecurity anomaly detection and investigation. It combines detection rules, policies, signatures IOC matching and event correlation with an Isolation Forest based behavioral anomaly model. Importantly the ML part is not asked to decide if something is an attack; instead it provides extra behavioral evidence that analysts can use with all other information. AegisML also connects events, detections, findings and investigations through relationships based on evidence. Includes MLOps practices like dataset management, experiment tracking, model versioning, registration, drift monitoring and controlled retraining.
In short AegisML bridges the gap, between security detection and ML lifecycle management by bringing evidence, analysis, investigation and model governance together in one system. Its main contribution is not an anomaly detection algorithm but a more connected, trackable and controlled way of doing cybersecurity anomaly detection.

Keywords: Cybersecurity, Anomaly Detection, MLOps, Security Logs, Isolation Forest, Evidence Driven Investigation, Security Analytics, Model Lifecycle, Drift Detection, Model Governance.

1.INTRODUCTION

Cybersecurity environments create a lot of data that keeps growing all the time. This data comes from things like authentication systems, applications, network hardware, security tools, operating systems and other services that are connected. These sources create types of logs and events. They do this at speeds and in different formats. For someone who works in security looking through all of this by hand is hard. Takes a lot of time. The main problem is not just collecting security data. Also finding the events that matter and understanding them properly.Traditional security monitoring still depends a lot on set rules, known patterns, policies, limits, signs of an attack and checking events. These methods are still important because the way they find things is clear and can be checked by a security person. When something known happens the reason

for a warning can usually be explained by the rule that started it.. There is a big problem with this approach. It mainly finds situations that have already been set. If a hacker changes how they act or does something that does not match a rule that action might not be found.Machine learning-based methods for finding things offer another way to look at this. Of relying completely on set conditions an anomaly detection model can learn patterns from the data and spot things that do not fit those patterns. Methods like Isolation Forest are especially helpful when it is hard to get examples of every attack. Previous studies on security and system logs have also shown how machine learning and deep learning can find behavior in log data.There is an important difference that needs to be clear between an unusual event and a confirmed security event. An anomaly detection model can show that something is different from what's expected but that does not mean it is bad. For instance an unusual login, system activity or a sudden change in how resources are used could have a real reason. If every strange thing is seen as an attack the number of warnings can go up and the person looking at them might not have the right information to decide. A cybersecurity system should use ML results as help not as the answer.Another problem comes after a machine learning model is built. In a real security setting the model can't just be trained once. Then left. How it works depends on the data used the features picked, the version of the model and the environment it is used in. Changes in how people act network activity, applications or how attacks happen can affect the model over time. This means there is a need for things like managing data, tracking experiments keeping track of versions of models watching for changes and retraining in a controlled way. These are part of the MLOps process.Even though these tools are available security detection and managing the life cycle of machine learning are often seen as separate. A security system might give a warning without showing the link to the model, data or experiment that caused the signal. Also an MLOps system might manage data and models without knowing the security evidence behind a detection. This split makes it harder to see how a result was made and to repeat or check the analysis later.To fix these problems this paper introduces AegisML, a Rule-First and Evidence-Driven MLOps Framework for Cybersecurity Anomaly Detection and Investigation. The framework mixes security detection with machine learning-based behavior analysis. Rule-based methods, like rules, signs, checking for attack signs and connecting events are still a part of the detection process. An Isolation Forest model adds another view of behavior. The two ways are seen as sources of security information not just one signal.AegisML also focuses on the proof that comes with a detection. Events, findings, risk details, investigation notes and machine learning results stay connected so that a person can look at a result in the way. Explainability is added through SHAP-based analysis to show what factors helped create an ML result. At the time the system keeps the difference between something the model found and the final security decision made during an investigation.The framework adds MLOps features into the security process. Managing data, tracking experiments keeping model versions registering models watching for changes finding differences and retraining in a controlled way are all part of the system. This helps keep a connection between the security work and the life cycle of the machine learning part.The main point of AegisML is to bring different ways of working. It does not offer a way to find strange things. Instead it offers a way to mix old security methods, behavior analysis looking at proof being clear about how things work and managing the life cycle of machine learning in one system. The goal is to make finding things in security easier to follow easier to understand and

2. LITERATURE REVIEW

2.1 Security Log Anomaly Detection

Security logs and system logs are sources of information for spotting unusual behaviour and looking into suspicious events. Traditional security monitoring relies heavily on rules, signatures, thresholds, policies and indicators of compromise. These approaches work well when the threat is known because the detection logic is clear and specific.. Behaviour that does not match any existing rule can go unnoticed. Machine learning-based anomaly detection offers a way to find issues by learning patterns from log data and flagging observations that deviate from normal behaviour. The DeepLog method showed that it is possible to learn patterns in system logs to detect anomalies. LogAnomaly went further by considering

both the sequence of events and quantitative data while LogBERT used transformer-based models to understand and detect log sequences.These studies show that machine learning can uncover patterns in logs that're hard to define using manual rules.. An anomaly score or label does not automatically mean an attack has happened. A security analyst still has to examine events understand the context and gather supporting evidence to judge how serious the anomaly really is.

2.2 Rule-Based and ML-Based Detection

Rule-based detection and machine learning-based detection offer kinds of insights. Rules are effective for detecting known threats with definitions. Machine learning methods on the hand can point to unusual behaviour that might not yet have a rule. Of replacing one approach with the other both can be used together during an investigation. They serve as sources of information.

This difference is important in AegisML. The machine learning part is meant to find activity and add evidence not decide on its own that an event is an attack.

2.3 Explainability and MLOps

Explainability is a concern when machine learning is used in security analysis. Security analysts often need to know why a certain log or event was flagged as anomalous. AegisML includes SHAP-based analysis to help explain the reasoning, behind machine learning outputs.

MLOps goes beyond building models. It involves managing datasets, tracking experiments handling model versions registering models, monitoring performance detecting data drift and retraining models. These practices are essential because security environments and the data they generate can change over time.

3. RESEARCH GAPS IN CYBERSECURITY ANOMALY DETECTION MLOPS

Gap 1: Separation Between Deterministic Detection and Behavioral Machine Learning.

Deterministic detection uses rules, policies, signatures, thresholds and indicators of compromise. Machine learning detection looks at deviations from learned behavior. These two parts are usually built separate. What is missing is an architecture that joins them properly: deterministic detection stays the security layer and machine learning adds complementary behavioral insight instead of competing with deterministic detection.

Gap 2: Disconnection Between Security Evidence and Machine Learning Lifecycle.

Security events, findings and investigation records normally exist apart from machine learning artifacts such as datasets, experiments and model versions. This separation makes it hard to trace a signal back to the specific model and data behind it. A unified framework must link security evidence directly to the machine learning component's life cycle.

Gap 3: Weak Separation Between Anomaly and Attack Determination.

An unsupervised anomaly detection model can say that something is unusual. It cannot say that it is malicious. Treating every anomaly as an attack causes confusion and erodes transparency. Therefore a cybersecurity framework must keep anomalies and security findings clearly separate.

Gap 4: Limited Lifecycle Management in Security Machine Learning Systems.

Many anomaly detection systems in security stop at data preparation, training and prediction. They leave features such as dataset versioning, experiment tracking, model versioning, monitoring, drift detection and controlled retraining underdeveloped. Proper management of the machine learning component, throughout its life requires these MLOps capabilities to be built in not added

Gap 5: Lack of an Integrated Security and MLOps Architecture.

Existing tools already perform detection, anomaly analysis, investigation or MLOps well on their own but rarely combine all these functions in one lightweight framework. What remains missing is an architecture that brings detection, behavioral analysis, security evidence and controlled machine learning life-cycle management into one unified system.

4. PROPOSED FRAMEWORK

AegisML starts where any of this has to start: collecting and preparing cybersecurity logs and events. That data is validated, normalized and pushed through feature engineering until it is fit for analysis.

From there it is analyzed two ways at deterministic mechanisms apply predefined rules, policies, signatures IOC matching and event correlation to catch known conditions while Isolation Forest [1] runs in parallel, to flag observations that deviate from learned behavior.

These outputs feed into security findings. The evidence that supports them. SHAP-based analysis [6] adds explainability on top ML generated anomaly information is easier to understand rather than just a number. Throughout AegisML keeps an anomaly signal and a confirmed security finding as two different things machine learning informs the decision it does not make it alone.

Around all of this sits the MLOps layer dataset management, experiment tracking, model versioning, registration, monitoring, drift detection and controlled retraining which gives the ML component real lifecycle management and keeps the data, experiments and models used in development and operation traceable.

5. SYSTEM ARCHITECTURE

The AegisML architecture, shown in Fig. 1 Shows a step-by-step process. First the system takes logs and events. Then the system runs both rule-based checks and machine-learning analysis. AegisML then produces findings. Manages the machine-learning life cycle. All of these parts—data work, security checks, explanations, model operations, deployment and ongoing improvement—are linked together in one system

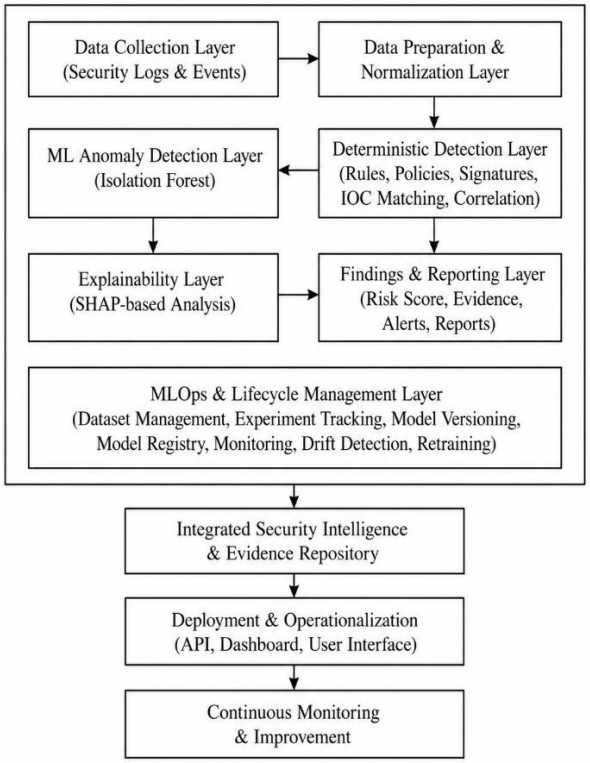

Fig. 1. System Architecture

The system starts with the Data Collection Layer. Here cybersecurity logs and events arrive. Next the Data Preparation and Normalization Layer turns data into a clean usable format. After cleaning the data flows into two layers. One layer is the Deterministic Detection Layer, where rules, policies, signatures IOC matching and correlation are applied. The other layer is the ML Anomaly Detection Layer, where Isolation Forest is used to spot patterns.

After that the Explainability Layer uses SHAP analysis to explain the machine-learning results. Then the Findings and Reporting Layer gathers risk details, evidence, alerts and reports. At the time the MLOps and Lifecycle Management Layer keeps track of datasets, experiments, model versions, the model registry, monitoring, drift detection and retraining.

The security intelligence and evidence created during the process are stored in the Integrated Security Intelligence and Evidence Repository. From this repository Deployment and Operationalization make everything available via APIs, dashboards and user interfaces. Continuous Monitoring and Improvement then keep the system and its models working over time.

6. ANALYSIS OF EXISTING SYSTEM CATEGORIES

- Rule-Based Detection Systems rely on predefined rules, signatures, policies, thresholds and indicators of compromise to identify known security conditions. Because the logic behind a detection can be clearly written out and reviewed directly these systems remain transparent. They are only as effective as the rules that drive them and a behavior that no one has written a rule for can simply go unnoticed.
- SIEM and Event Correlation Systems gather security events from sources into centralized monitoring and alerting [10]. Correlating events this way highlights connections that improve visibility into what's actually happening but these systems are designed for event collection, correlation and monitoring not for running a complete machine learning lifecycle behind behavioral anomaly detection.
- ML Based Log Anomaly Detection Systems use machine learning to identify patterns, in logs and event data that predefined rules would miss [2][4]. That behavioral detection capability is truly helpful. An anomaly flagged by an ML model is not automatically a confirmed attack and the ML component underneath still requires proper data preparation, model management, monitoring and lifecycle control.
- MLOps Platforms manage the machine learning workflow itself dataset management, experiment tracking, model versioning, registration, monitoring and retraining [7]. They make ML more reproducible and easier to manage over time. General purpose MLOps platforms

System Category	Rule-Based Detection	Behavioral Anomaly Detection	Evidence ntegration	Explainabiliy	MLOps Lifecycle	Investigation Support
Rule-Based Detection Systems	Strong	Limited	Limited	High	Limited	Moderate
SIEM & Event Correlation Systems	Strong	Limited	Strong	Moderate	Limited	Strong
ML-Based Log Anomaly Detection	Limited	Strong	Limited	Varies	Limited	Moderate
General MLOps Platforms	Not a primary focus	Model-dependent	Limited	Model-dependent	Strong	Limited
AegisML	Strong	Strong	Strong	Integrated	Strong	Strong

Table 1. Comparison of Existing System Categories with AegisML

The comparison shows that current cybersecurity systems usually focus on parts of detection and investigation instead of offering a complete solution. Rule-based detection systems work well for finding known and set rules-based events. They don't do much for detecting unusual behavior or managing the machine learning lifecycle. SIEM and event correlation systems offer support for combining evidence and investigating but analyzing behavior and MLOps features are not always the main focus. Machine learning-based log anomaly detection helps find patterns but connecting these findings with security evidence and investigation steps can be hard. General MLOps platforms are good at managing models over time. They are not made specifically for security investigations.AegisML is built to bring all these abilities together in one place. It uses rule-based detection along with behavior analysis links detection results with security evidence explains findings, from machine learning and uses MLOps methods for keeping track of model versions watching them and making sure updates are done properly. This combined setup is meant to help analysts during the detection and investigation process while making sure security findings can be tracked.

7. DESIGN PRINCIPLES OF THE PROPOSED FRAMEWORK

- **Security-Centric Integration:** The design considers security as a central aspect of AegisML. The components of ML and MLOps are incorporated into the security lifecycle, and not kept isolated. Hence, the process of analysing the event. The handling of the model that analyses it are intrinsically tied into the security environment.
- **Layered Detection Design:** Layered Detection Design proposes the notion of having two different levels of detection. One aims at using rule, policy, signature IOC matching and event correlation to identify known events. The other level employs ML to detect abnormal behaviours. The layered approach aims at providing a detection view that helps the analyst distinguish what each level contributes.
- **Evidence-Centered Output:** Evidence-Centered Output assumes that a detection is more valuable when the analyst can understand the evidence that led to the identification. AegisML provides a way to associate a finding to events, findings, risk information and other supporting evidence. It aids the analyst to retrace the information that led to a particular detection.
- **Controlled ML Integration:**

Controlled ML Integration posits that the ML model should not be tasked to make a determination on whether an event is malicious or not. Rather, it should be used for identifying abnormal behavioural patterns and provide additional context for investigation. The analyst uses the information, along with security-related evidence and determine if there is an underlying concern. The framework looks at how the ML component can be placed within the cybersecurity ecosystem and the considerations for its evolution. It looks at dataset curation, experimentation, model versioning, registry, monitoring, drift detection and retraining as ongoing aspects that influence the model's ability to perform alongside the evolving needs.

8. FUTURE RESEARCH DIRECTIONS

- While AegisML allows for detecting cyber anomalies analyzing evidence interpreting findings and managing the MLOps lifecycle, there are plenty of after-sale support issues to consider in the future. First of all, there is a need to consolidate heterogeneous security data across different areas, including network traffic, host activity, identity data, application logs, and cloud security resources, among others, into a unified structure to facilitate understanding the context of behaviors detected across the environment. This way, the detection of anomalous patterns will become more accurate, with the ability to leverage various sources of information to obtain in-depth insights about the system under control.
- Another research direction is to consider temporal and sequential analysis to discover patterns that involve interdependent events. Such patterns are critical to identifying long-term coordinated attacks that may evolve through multiple stages before targeting specific systems, resources, or data. Finally, multiple anomaly detection techniques, including autoencoders, clustering, ensemble, sequence models, and others, can be considered to support the main method based on Isolation Forest utilized in the framework.
- In addition, future research should explore analyst-in-the-loop learning approaches, where cybersecurity analysts provide feedback to the ML model about the true-positive detected anomalies or incidents, thus refining the training data and contributing to the continuous improvement of the algorithm. In this way, the framework can incorporate threat intelligence data to enrich the information about anomalous activity detected by the model, aligning it with known indicators of compromise, threats, and behaviors from external databases and improving detection accuracy. Moreover, future iterations of AegisML can enhance evidence visualization and interpretation, offering more extensive insights into the detected anomalies by linking them to relevant behaviors, evidence, characteristics, and characteristics that contributed to the suspicion, thus presenting a more comprehensive view of security incidents.
- Finally, the framework's future development should include the enhanced model governance component, which would add data validation, model certification, deployment pipelines, model monitoring, drift detection, and retraining capabilities. This way, AegisML will evolve into a complete ML-based platform that ensures effective operation within an organization, offering comprehensive analytics, detecting, and investigating anomalous activity while maintaining proper documentation and human oversight.

9. CONCLUSION

AegisML is a rule-based evidence-focused MLOps framework, which helps to detect and investigate cybersecurity anomalies. AegisML combines the established security rules and machine learning which looks out for behavior. This provides insight and AegisML keeps the raw signal of an anomaly from a confirmed security incident. Cybersecurity environments are prone to generating numerous security events, which often complicate the process of distinguishing genuine anomalies and responding to them in a timely manner. I perceive that AegisML framework presents a paradigm for addressing such issues, as the proposed solution effectively combines rule-based and behavioral anomaly detection within the context of a MLOps architecture. In place of relying on single detection mechanisms, which pose risks of false positives and inadequate response procedures, the AegisML framework utilizes machine-learning predictions as supportive kevidence, alongside well-defined rules, security context and human expertise.

The AegisML framework was additionally designed to incorporate anomaly detection into explainability-first and MLOps-aware contexts. Such design considerations as SHAP-based interpretability facilitate the identification of influential factors on a case-by-case basis, whereas model versioning, monitoring, drift detection and controlled retraining offer traceability and lifecycle management throughout the operational phase. By standardizing the cybersecurity-specific investigation procedures, which are traditionally executed by humans, the AegisML framework establishes a synergy between the domains of cybersecurity investigation and ML operations.

Overall, AegisML framework was evaluated as a viable architecture and methodological approach to improving traceability, explainability and operational control within the context of cybersecurity-oriented anomaly detection. As a concept, the AegisML framework adheres to the rule-first paradigm and incorporates evidence-based analysis, human-in-loop security practices and standardized MLOps-aware lifecycle management to standardize the implementation of security-focused analytics in a manner that supports human decision-making. Future research directions may build upon the findings of this study to expand the capabilities of the AegisML framework, potentially by implementing multi-source telemetry, analysis, more sophisticated anomaly detection methodologies, threat intelligence-aware approaches and analyst-in-the-loop learning mechanisms.

REFERENCES

[1] F. T. Liu, K. M. Ting and Z.-H. Zhou, "Isolation Forest " in Proceedings of the Eighth IEEE International Conference on Data Mining (ICDM) pp. 413–422, 2008. (Monash University)

[2] M. Du, F. Li, G. Zheng and V. Srikumar "DeepLog: Anomaly Detection and Diagnosis from System Logs through Deep Learning " in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 1285–1298, 2017.

[3] W. Meng, Y. Liu, Y. Zhu, S. Zhang, D. Pei Y. Liu, Y. Chen, R. Zhang, S. Tao, P. Sun and R. Zhou, "LogAnomaly: Unsupervised Detection of Sequential and Quantitative Anomalies in Unstructured Logs " in Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI) pp. 4739–4745, 2019. (IJCAI)

[4] H. Guo, S. Yuan and X. Wu, "LogBERT: Log Anomaly Detection via BERT " in Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN) pp. 1–8, 2021.

[5] M. Landauer, S. Onder, F. Skopik and M. Wurzenberger "Deep Learning for Anomaly Detection in Log Data: A Survey," Machine Learning with Applications vol. 12 P. 100470 2023. (DOI)

[6] S. M. Lundberg and S.-I. Lee "A Approach to Interpreting Model Predictions " in Advances in Neural Information Processing Systems 30 (NeurIPS) pp. 4765–4774, 2017. (ML Anthology)

[7] D. Kreuzberger, N. Kühl and S. Hirschl "Machine Learning Operations (MLOps): Overview, Definition and Architecture " IEEE Access, vol. 11 Pp. 31866–31879 2023.

[8] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary M. Young, J.-F. Crespo and D. Dennison "Hidden Technical Debt in Machine Learning Systems " in Advances in Neural Information Processing Systems 28 (NeurIPS) pp. 2503–2511, 2015. (ML Anthology)

[9] E. Breck, S. Cai, E. Nielsen, M. Salib and D. Sculley "The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction " in Proceedings of the 2017 IEEE International Conference on Big Data pp. 1123–1132 2017.

[10] G. González-Granadillo, S. González-Zarzosa and R. Diaz "Security Information and Event Management (SIEM): Analysis, Trends and Usage, in Critical Infrastructures " Sensors, vol. 21, No. 14 P. 4759 2021. (MDPI)