Power System Technology
ISSN:1000-3673

Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

Adaptive Api Security with Behavior-Based Machine Learning Models in Mulesoft

Rakesh konda

Independent Researcher

MuleSoft Developer

Email: konda9406@gmail.com

Abstract - Adaptive security is a security approach that is utilised to respond to major cyber threats in real time. This paper investigates the application of behaviour-based ML or machine learning frameworks into MuleSoft's API security model to address certain limitations of static security systems. This paper critically analyses the existing limitations in native security processes of MuleSoft, identifies major ML methods for anomaly detection, including Autoencoders and Isolation Forest, and investigates data privacy, model drift, and others as threats to deployment. This paper integrates a mixed research method with both secondary quantitative and qualitative methods, which leads to an adaptive and flexible security model. Furthermore, behaviour-based machine learning in MuleSoft's API design, Daily retraining cycles, and others were included as recommendations in this paper.

Index Terms- API, Behaviour-Based Machine Learning Models, API Security, MuleSoft

I. INTRODUCTION

A. Background to the Study

Adaptive API security is a dynamic context that adjusts security measures in real-time based on analysis of API usage and actions, rather than relying on static norms. Organisations are adapting APIs for faster data exchange, and security in API has become a major concern, specifically in distributed and dynamic landscapes. The incorporation of APIs in the current software landscapes has typified the vulnerability of APIs in the face of cybersecurity risks [1]. Although MuleSoft helps speed up API creation, it is still vulnerable to modern, advanced security threats.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

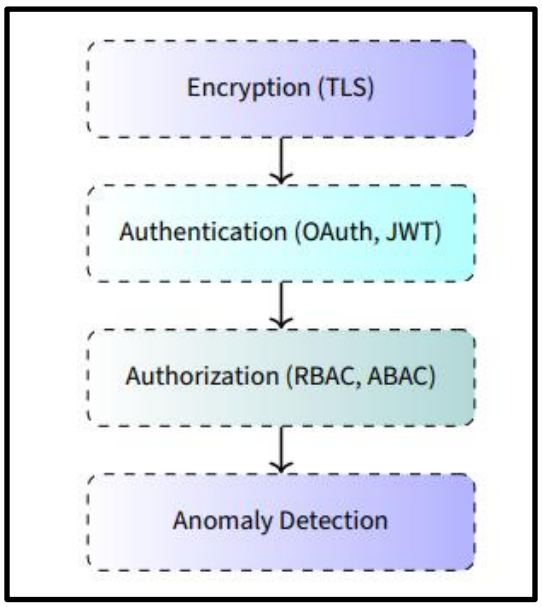

Figure 1: API Layers

[3]

Figure 1 highlights API layers including "Authentication, Authorization, and Anomaly Detection." Older security strategies find it problematic to find anomalies and deal with zero-day threats. Applying behaviour-based machine learning models can be used to highlight suspicious activity and take action to address threats in real time. This paper investigates how ML is being used in MuleSoft to make security systems more reliable and proactive in facing emerging hazards in digital infrastructure.

B. Overview

This research explores how "behaviour-oriented machine frameworks" can be applied to MuleSoft's API modelling to create an adaptive and flexible security mechanism. This shows that earlier static rules no longer work effectively as the threats in cybersecurity have become more dynamic. An adaptive approach for real-time detecting malware using API highlight sequences invoked by the target program [2]. Various machine learning techniques for detecting anomalies are listed by the study, such as Isolation Forest, Autoencoders, One-Class SVM, and others, and their suitability for use with MuleSoft is considered. Various case study examples show how threat detection has improved, and there are fewer false positives. This paper bridges practice and theory by suggesting a reliable, intelligent, and scalable layer of security that evolves with the user's patterns of the system and the user.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

C. Problem Statement

Apart from the strong API management abilities, it lacks address validation for dynamic, thriving, and behaviour-oriented security processes, making the APIs vulnerable to enlightened attacks, including insider threats and API abuse. API security in enterprise-level distributed systems, where many services connect and numerous data exchanges occur [3]. Older security methods respond to issues after they occur and are not flexible enough to adapt right away. This study considers the main threat by suggesting how machine learning models that focus on behaviour can be used with MuleSoft to ensure better security. As a result, the research improves the threat detection on the platform and adds to the development of intelligent cybersecurity, since the framework adapts to the changing environment and threats.

D. Objectives

The primary goals of this study are: 1. To highlight the current limitations of MuleSoft's native API security systems. 2. To explore ML techniques effective for behaviour-based anomaly detection in API traffic. 3. To refer to deployment threats of behaviour-based machine learning frameworks, including data privacy, computational overhead, and model drift. 4. To assess the accuracy and performance of the behaviour-based security framework to highlight challenges. These highlighted research objectives aim to explore, create, and evaluate a flexible security model for MuleSoft APIs through behaviour-based machine learning frameworks to highlight and mitigate anomalous and malicious observations in real time.

E. Scope and Significance

This paper concentrates on improving MuleSoft's API security by applying behaviour-based machine learning frameworks able to highlight anomalies in real time. Thus, the scope of this paper revolves around the evaluation of preferred ML algorithms, creating flexible detection frameworks, and applying them insight the gateway of MuleSoft. Additionally, the significance of this research lies in modifying major API defence processes apart from the traditional rule-oriented processes, particularly aimed at increasing API-oriented cyber challenges. API tracers that can be used even for security research [4]. Hence, by referring to threats such as data privacy, model drift, and others, this paper became pivotal to creating an intelligent and resilient security model for companies depending on MuleSoft for data exchange and digitalisation.

II. LITERATURE REVIEW

A. Current Limitations of MuleSoft's Native API Security Mechanisms

MuleSoft creates reinforcement in the API management tools, such as policies for rate limiting, access control, and authentication. Nevertheless, its core security model is based on unchanging rules, which cannot correlate with the current threat environment. Pre-defined rules are effective for specific actions, though they find it problematic to detect new types of threats, abuse of APIs, or sneaky behaviour by insiders. Thus, APIs can be thought of as a coordinator that bridges communication between multiple platforms and systems [5]. For example, a financial services firm

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

using MuleSoft experiences challenges when malicious bots pose as genuine users to gather their data. The security system did not highlight unusual actions, since the suspicious behaviour was not above the set limits.

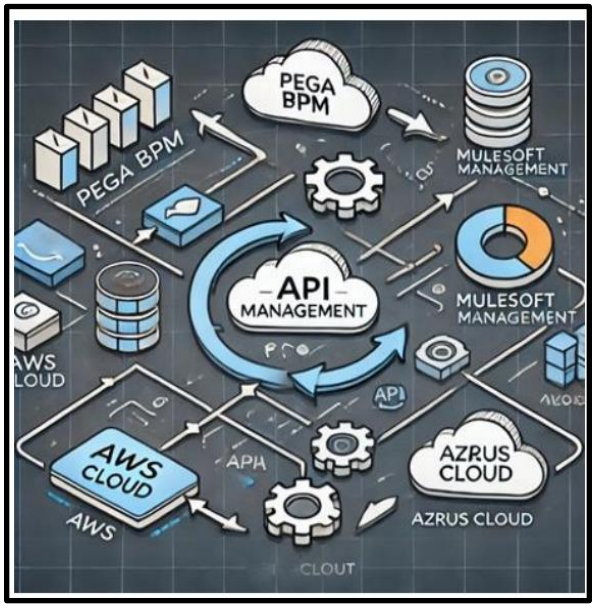

Figure 2: API Management

[5]

This highlights that following such models tends to lead them to report many false positives or overlook important details. As per the above figure, API management includes parameters such as AWS Cloud, PEGA BPM, AZRUS Cloud, and others [5]. This shows that such mechanisms need to be able to learn and react to the behaviours of users all the time. Using machine learning, one can highlight unexpected changes in API activities proactively.

B. ML Techniques Suitable for Behaviour-Based Anomaly Detection

"Behaviour-based anomaly detection" highlights ML to observe deviations from existing patterns in the API integration. Isolation Forest is preferred as it individually separates data points and has been found suitable for traffic data from APIs. With the help of deep learning, autoencoders can highlight normal actions from unusual ones based on reconstruction errors. OC-SVM is used to investigate "language model-based feature types" such as "attributed token Grams" [6]. One-Class Support Vector Machines (SVM) models the distribution of normal actions, which is also impactful for anomaly detection in network processes, however, these can suffer from scalability threats with wider datasets. Most works in the field observed that unsupervised and semi-supervised techniques are better, as labelled attack data is not always easily available. API platforms combine with these models to stay up-to-date by noticing shifting patterns in traffic.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

C. Deployment Challenges

Using machine learning models in API security, such as MuleSoft, poses several threats to the system. As the API has a different usage, as a result, the model's accuracy can decrease over time. Thus, to avoid this, models need to be updated all the time, and real-time monitoring is required. This is particularly important to consider data privacy during the analysis of sensitive types of data, especially under attributes such as the GDPR or HIPAA. Security tools created to facilitate developers in managing vulnerabilities may negatively impact developers' security knowledge instead of aiding it [7]. Computational overhead has been identified as another challenge, as ML frameworks need to process "high-velocity API traffic" without acknowledging latency. This can be necessary to use small models on the edge for deployment in many places. Considering these issues specifies that the adaptive security solution is effective, follows the rules, and is pivotal in businesses.

D. Assessing Performance and Accuracy of the Security Framework

Assessing the efficacy of behaviour-based ML frameworks includes measuring core elements, including false-positive rate, recall, and precision, with the help of API traffic in the real-world context. Incorporating the Receiver Operating Characteristic (ROC) Curve theory helps to highlight the relationship between specificity and sensitivity, and sets the right standard for decision-making. The ROC approach can depict the trade-off between benefits, such as the "true positive rate", and costs, such as the "false positive rate" [8]. Additionally, APIs serve as major connectors in contemporary finance [9]. Observing how real-world applications use the system is necessary to check its stability in multiple settings. Adaptability to new risks and still keeping legitimate users unharmed is a core reason for the model's success. Regular testing against standard data and test environments guarantees that the model can work seamlessly and be ready for use in the workplace.

III. METHODOLOGY

A. Research Design

Research design in methodology has been identified as a systematic plan or model that describes how research will be executed. Thus, an "**explanatory research design**" has been chosen as a research design in this paper to explore, create, and evaluate a flexible security model for MuleSoft APIs through behaviour-based machine learning frameworks. An explanatory design was integrated to combine and mix multiple datasets to be collected and interpreted [10]. Explanatory design is effective for achieving the goal by connecting behaviour-based machine learning with improved results in MuleSoft API security. An analysis of how ML methods impact threat detection leads the study to review the system's performance, issues, and flexibility with regard to research purposes.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

B. Data Collection

This study of Adaptive API Security employs a multi-methods research approach, using both "**secondary quantitative and qualitative techniques**". Qualitative methods utilise various empirical materials, including case studies, life experiences, and stories [11]. Data sources used for the secondary qualitative research are academic articles, case study examples, and industry reports. After that, statistical charts, graphs, and metrics are collected and further interpreted in a secondary quantitative method. Thus, the incorporation of mixed research methodology improved reliability with the help of "data triangulation" and increased validity by creating comprehensive overviews, cross-verifying study outcomes, and navigating evidence-oriented conclusions from multiple trustworthy data sources.

C. Case Studies/Examples

Case Study 1: Global API Ecosystem with MuleSoft

Siemens integrated with MuleSoft to support a system of APIs that is safe and scalable across its business. With the help of API policies such as throttling and OAuth2, they ensured that their security adapts to every partner and team member [12]. This measure made it easier to detect and stop anything strange happening on the networks.

Case Study 2: Accelerates Digital Transformation Using MuleSoft

MuleSoft is used by Unilever to safely connect its previous systems to new cloud solutions and accelerate e-commerce innovation [13]. They used permissions for users, defined several security policies, and analysed API activity to find irregular usage patterns.

Case Study 3: Modernises Legacy Systems with MuleSoft's Platform

Airbus ensured efficient API security and updated its old technology systems with the Any point Platform from MuleSoft. This company incorporated an enterprise-grade API platform to enable digital transformation at scale [14]. Real-time observation enabled them to adapt access according to the actions of users, an approach toward being behaviour-based.

D. Evaluation Metrics

False-positive rate, F1-score, recall, and precision have been identified as some evaluation metrics included in this research. These research-oriented evaluation metrics specify validation of research objectives, navigate model tuning, and decrease misclassification, improving performance, reliability, and real-time deployment for a proactive API security model.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

IV. RESULTS

A. Data Presentation

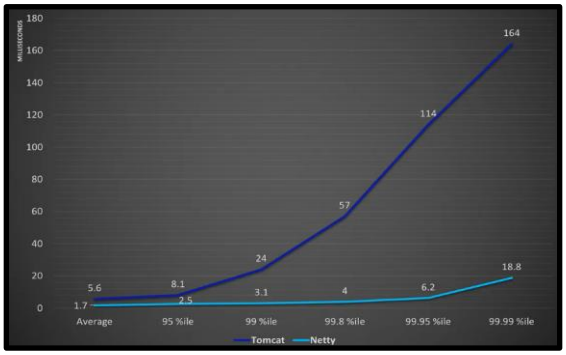

Figure 3: API Gateway- Latency comparison

[15]

Figure 3 demonstrates is showing a comparison of API Gateway latency while functioning as HTTP containers. Analysts review the usual latency as well as different high percentile marks [15]. Netty shows much lower latency at every point, and the difference in performance increases a lot as the latency becomes extremely high. From these findings, Netty is better at maintaining low answer times, apart from the user counts.

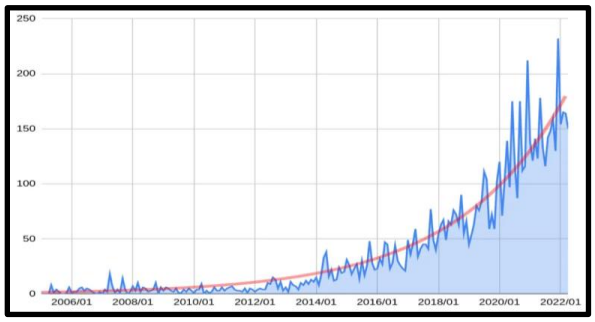

Figure 4: API Exploits and Monthly Issues Reported

[16]

Figure 4 shows that the cases of new API exploits and security problems have been going up steadily since 2006. In the beginning, the threat was very low, but since 2016, it has increased significantly, reaching more than 200 per month around 2020 to 2022 [16]. The upward trendline highlighted in red demonstrates that APIs are used more and more and, at the same time, cause more vulnerabilities. MuleSoft and such platforms require improved security methods that can address current threats as they occur.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

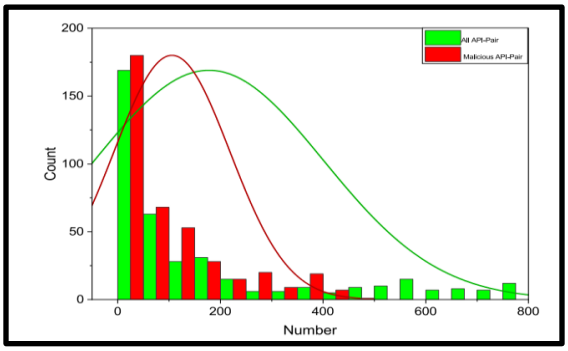

Figure 5: Statistics of API number

[2]

The histogram plots data showing "the count of malicious API-Pairs" is larger in comparison to "API-Pairs in the API-Pair number" less than 400 [2]. Apart from that, all the statistics related to "malicious API-Pair" were less than 500 [2]. Additionally, tests of the API's behaviour validate the incorporation of functional and non-functional requirements [17].

B. Findings

From Figure 3, it was found that Netty gets better API Gateway response times than Tomcat. At the highest percentile. Netty is very effective in tackling huge amounts of traffic and ensures it responds with faster, more reliable times, mainly for critical, rare, or tricky requests [15]. If an organisation wants to add behaviour-based ML security to MuleSoft, they need to consider the additional time it takes, since speedy threat detection and quick responses are key in some cases [15]. Thus, making the models as light as possible and improving performance is very important for achieving both safety and working performance on MuleSoft. Data in Figure 4 makes it clear that API-related security risks have increased significantly over the past decade, highlighting how vulnerable the current situation is [16]. This shows that rule-based systems are not enough and that security systems need to be able to change with new threats. If API management platforms such as MuleSoft use behaviour-based machine learning, they can become more resistant to abnormal traffic and fresh threats. The outcome shown in Figure 5 indicates that "Basic Sequential Algorithmic Scheme or BSAS clustering" had a positive impact on decreasing the amount of data that needed to be highlighted.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

C. Case Study Outcomes

Case Study Name	Case Study Company	Case Study Outcome	Relevance to Current Research
Siemens Builds a Global API Ecosystem with MuleSoft	Siemens	Implemented a secure, scalable API framework using OAuth2 and throttling policies to monitor and restrict access [12].	Highlights the role of layered API security and monitoring, aligning with ML-driven adaptive anomaly detection.
Unilever Accelerates Digital Transformation Using MuleSoft	Unilever	Connected legacy and cloud systems with role-based access control and analytics to detect unusual API behaviour [13].	Reinforces the importance of behavioural analytics and access control, key pillars of behaviour-based security models.
Airbus Modernises Legacy Systems with MuleSoft's Platform	Airbus	Adopted real-time API monitoring and adaptive access management based on usage patterns during modernisation [14].	Demonstrates the transition toward behaviour-aware API management, setting a foundation for ML-based anomaly detection.

Table 1: Case Study Outcome

[Source: Self-Created]

The case study examples in Table 1 have highlighted how MuleSoft-based API security initiatives navigate behaviour-based and flexible frameworks coordinated with ML-based detection of anomalies.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

D. Comparative Analysis

Author	Aim	Findings	Gaps identified
[5]	This paper aims to highlight the application of "PEGA and MuleSoft with Cloud Services."	Organisations can leverage PEGA to automate workflows and improve customer service experiences.	Lack of primary research
[6]	This paper aims to highlight "anomalies using k-nearest neighbours."	There are three feature extraction methods, such as "token, temporal token, as well as attributed token grams," and these can be used to embed semi-structured Information contained in security audit log data.	Lack of critical analysis of outcomes
[7]	This article aims to "highlight Adaptive Security Interventions."	As per the analysis, "adaptive security interventions" act as a mitigation that responds to the changing security demands of individual developers.	Less empirical validation for security mitigations and measures
[9]	This paper aims to identify the role of "API Security Through Genetic Algorithm-based Machine Learning Model in FinTech."	API security is robust as it maintains the credibility and trustworthiness of financial services in light of the ongoing expansion of the digital economy.	Lack of risk management analysis

Table 2: Comparative Analysis of Literature Review Sources

[Source: Self-Created]

Comparative interpretations in the above table help to fulfil research aims and objectives by identifying aims, trends, and gaps, specifying refined knowledge of the future of Adaptive API Security with Behaviour-Based Machine Learning Models.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

V. DISCUSSION

A. Interpretation of Results

Methods have collective support to fulfil the parameters of the research objectives, such as the graphs relating to latency and API exploits outlining how MuleSoft is struggling and gaining more attack risks, as expected for the first objective. Cultivation of isolation Forest, Autoencoders, and One-Class SVM to evaluate effective models was done in this paper to fulfil the parameters in the second research objective. Threats such as model drift and difficulties with resource use were highlighted as part of the third objective. Siemens, Unilever, and Airbus's case studies highlight that adaptive API security is used in practice and keeps evolving, in line with the last research objective.

B. Practical Implications

This paper creates major practical value for companies that depend on APIs for digital activities. When MuleSoft uses behaviour-based models, companies are better equipped to find and prevent unusual traffic, which reduces the chances of data breaches and problems with the service [18]. Such technologies also help organisations follow new cybersecurity rules by alerting them in real-time and providing the appropriate reaction to threats. As a result, this method also increases MuleSoft's ability to handle many workloads and withstand mistakes. These findings can be used by organisations to design their API infrastructure to support automation, intelligence, and security, and still maintain their usual operational efficiency and performance.

C. Challenges and Limitations

This paper has multiple major limitations and challenges. For example, using secondary data and relying on qualitative information was not enough to verify the connection between theory and practice. Model drift is an issue because, as APIs change, the model might lose accuracy, which often requires the model to be retrained. Additionally, making ML models that are based on behaviour introduces additional demands on the system's resources. This has become problematic to use all the payload information for training due to data privacy rules. Due to fewer case studies and research on MuleSoft API security using ML, it is difficult to test the results. Thus, these areas indicate room for further primary research in this field.

D. Recommendations

Companies can use behaviour-based machine learning in MuleSoft’s API design to improve their ability to detect threats easily. Daily retraining cycles can be applied to hold model drift as well, and lightweight algorithms need to be valued to decrease algorithmic overhead [19]. As part of ensuring privacy compliance, it is important to anonymise data before doing any other operations. The security department can use machine learning techniques together with rule-based approaches as a set of “set of IF-THEN rules” to improve security [20]. Additionally, user behaviour monitoring and fast dashboards help improve the way an organisation deals with API attacks.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

VI. CONCLUSION AND FUTURE WORK

This paper shows the increasing significance of behaviour-oriented and adaptive ML frameworks in increasing API security within the domain, such as MuleSoft. Using Isolation Forest and Autoencoders algorithms, firms can quickly observe unusual patterns in their API activities and provide better protection than traditional security rules. From the study, it is significant to highlight the importance of testing, confidentiality, and how the system can grow as needed. Further research needs to be attentive to creating privacy-preserving and lightweight ML frameworks optimised for deployment in manufacturing. Testing in different fields will help the generalisability of the research. As a result, using AI (XAI) tools will make it easier for people to trust and understand the decisions taken by automated security systems, helping AI become more involved in cybersecurity.

VII. REFERENCE LIST

[1] Ranjan, P. and Dahiya, S., 2021. Advanced threat detection in api security: Leveraging machine learning algorithms. International Journal of Communication Networks and Information Security, 13(1).

[2] Yang, S., Li, S., Chen, W. and Liu, Y., 2020. A real-time and adaptive-learning malware detection method based on api-pair graph. IEEE Access, 8, pp.208120-208135.

[3] Kaul, D. and Khurana, R., 2021. AI to detect and mitigate security vulnerabilities in APIs: encryption, authentication, and anomaly detection in enterprise-level distributed systems. Eigenpub Review of Science and Technology, 5(1), pp.34-62.

[4] D'Elia, D.C., Nicchi, S., Mariani, M., Marini, M. and Palmaro, F., 2021. Designing robust API monitoring solutions. IEEE Transactions on Dependable and Secure Computing, 20(1), pp.392-406.

[5] Singasani, T.R., 2020. Integrating PEGA and MuleSoft with cloud Services: Challenges and opportunities in modern enterprises. Journal of Scientific and Engineering Research, 7(3), pp.328-333.

[6] Duessel, P., Luo, S., Flegel, U., Dietrich, S. and Meier, M., 2020, May. Tracing privilege misuse through behavioural anomaly detection in geometric spaces. In 2020 13th International Conference on Systematic Approaches to Digital Forensic Engineering (SADFE) (pp. 22-31). IEEE.

[7] Rauf, I., Petre, M., Tun, T., Lopez, T., Lunn, P., Van Der Linden, D., Towse, J., Sharp, H., Levine, M., Rashid, A. and Nuseibeh, B., 2021. The case for adaptive security interventions. ACM Transactions on Software Engineering and Methodology (TOSEM), 31(1), pp.1-52.

[8] Han, H., 2022. The utility of receiver operating characteristic curve in educational assessment: Performance prediction. Mathematics, 10(9), p.1493.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

[9] Dhaiya, S., Pandey, B.K., Adusumilli, S.B.K. and Avacharmal, R., 2021. Optimizing API Security in FinTech Through Genetic Algorithm based Machine Learning Model. International Journal of Computer Network and Information Security, 13, p.24.

[10] Othman, S., Steen, M. and Fleet, J., 2020. A sequential explanatory mixed methods study design: An example of how to integrate data in a midwifery research project. Journal of Nursing Education and Practice, 11(2), pp.75-89.

[11] Taherdoost, H., 2022. What are different research approaches? Comprehensive review of qualitative, quantitative, and mixed method research, their applications, types, and limitations. Journal of Management Science & Engineering Research, 5(1), pp.53-63.

[12] Mulesoft.com, 2023. Siemens transforms how the world consumes energy using APIs, https://www.mulesoft.com/case-studies/api/siemens [Accessed on: 5th February, 2024]

[13] Mulesoft.com, 2018. Unilever accelerates eCommerce innovation using APIs, https://www.mulesoft.com/case-studies/api/unilever [Accessed on: 7th February, 2024]

[14] Mulesoft.com, 2018. Airbus' digital transformation takes flight with APIs, https://www.mulesoft.com/case-studies/api/integration-airbus [Accessed on: 11th February, 2024]

[15] Medium.com, 2018. API and Microservices Management Benchmarkv2.0, https://medium.com/paypal-tech/principles-of-an-api-gateway-2c18849ab3e2 [Accessed on: 19th January, 2024]

[16] Lab.wallarm.com, 2022. Evolution of API Security – A Practical Guide to Addressing API Threats in 2023, https://lab.wallarm.com/evolution-of-api-security-in-2023-a-practical-guide/ [Accessed on: 25th January, 2024]

[17] Bondel, G., Landgraf, A. and Matthes, F., 2021, July. API management patterns for public, partner, and group web API initiatives with a focus on collaboration. In Proceedings of the 26th European Conference on Pattern Languages of Programs (pp. 1-17).

[18] Chatterjee, A., Gerdes, M.W., Khatiwada, P. and Prinz, A., 2022. Sftsdh: Applying Spring security framework with TSD-based oauth2 to protect microservice architecture apis. IEEE Access, 10, pp.41914-41934.

[18] P. Chintale, R. K. Malviya, N. B. Merla, P. P. G. Chinna, G. Desaboyina and T. A. R. Sure, "Levy Flight Osprey Optimization Algorithm for Task Scheduling in Cloud Computing," 2024 International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS), Hassan, India, 2024, pp. 1-5, doi: 10.1109/IACIS61494.2024.10721633.

[19] Bucha, S. DESIGN AND IMPLEMENTATION OF AN AI-POWERED SHIPPING TRACKING SYSTEM FOR E-COMMERCE PLATFORMS.

Power System Technology
ISSN:1000-3673
Received: 06-08-2024 Revised: 15-09-2024 Accepted: 10-10-2024

[20] Yugandhar, M. B. D. (2023). Automate Social Sharing with Meta platform, Google feed, Linkedin feed, Google News, Fb, Instagram, Twitter. International Journal of Information and Electronics Engineering, 13(4), 7-15.

[21] Zhao, P., Cai, L.W. and Zhou, Z.H., 2020. Handling concept drift via model reuse. Machine learning, 109, pp.533-568.

[22] Liu, Q., Hagenmeyer, V. and Keller, H.B., 2021. A review of rule learning-based intrusion detection systems and their prospects in smart grids. Ieee Access, 9, pp.57542-57564.

[23] Venna, S. R. (2024). Leveraging Cloud-Based Solutions for Regulatory Submissions: A Game Changer. Available at SSRN 5283294.