Real-Time Security Alerts Automation and Orchestration using Wazuh and n8n

1st Mahnoor Shafi
Department of Information & Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
mahnoorshafi88@gmail.com

2nd Sana Tariq
Department of Information & Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
sana.tariq@iub.edu.pk

3rd Aoun Muhammad
Department of Information & Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
aoun.muhammad@iub.edu.pk

***Abstract*—In today's world, the frequency of cyber attacks is increasing day by day, and many open-source tools are used for threat detection and security in every organization. Many SIEM software, such as Wazuh, are used to detect threats and provide details about the attack, while n8n can serves like an automated tool. This research proposed an automated notification workflow design for incident response. It shows that, instead of manual monitoring, this integration can delivers real-time notifications to administrators via Gmail. This framework shows how to integrate Wazuh with n8n to produce real-time Gmail alerts. In this, the OAuth 2.0 protocol is used for a secure connection. In this, the original password cannot be used; the system was implemented with API credentials like Client ID and Client Secret tokens instead of the original password to maintain a secure connection. When Wazuh sends logs to n8n via webhook, they provide data in JSON format. Additionally, it works like a rule based system in the form of automation, in which if the attack reaches level 10+ or high severity, the system sends a Gmail alert; otherwise, Wazuh continues to detect every threat.**

***Index Terms*—Wazuh, n8n, SOAR, OAuth 2.0, Gmail alerts, Intrusion Detection, Automated Alerting.**

I. Introduction

The rapid growth of digital assets is increasing the cyber threats. It is a very serious problem to manage all threats simultaneously, and it is also very difficult to identify every single threat that is critical or not. Almost every organization establishes a security lab to secure its machines and conduct campaigns within its organization for threat detection and to make its system secure.

Traditional security systems totally depend on manual monitoring,where the administrator actively reveiw the dashboard, they can identify the threats; it takes a lot of time. Tools like Splunk and IBM QRadar are very expensive for small companies and students due to their high costs, and the technical setup and maintenance are also very complex; that's why an open source tool like Wazuh is very useful and also budget-friendly because it offers a detection system without any cost.

Existing solutions show that Wazuh, is the alternative of expensive tools, also they talk about the Telegram and Slack alerts in this they can use very high scripting but with the use of SOAR automation can make very easy process [1]. In today's world, OAuth has been implemented in other fields for security, but in this paper, we utilized it specifically to combine it with a SIEM tool and automation platforms for security; many attackers can find loopholes to compromise the system because in their system static password are used. In contrast, in this system, with the use of this OAuth protocol, it is more secure than other API systems because it is a token-based communication. There is very limited work on Wazuh and n8n integration the researchers show with the use of n8n they can reducing time from hours to seconds.

This framework shows that Wazuh and n8n integration can eliminate manual monitoring. In this integration, there is no need for heavy scripting. In this, Wazuh is used to collect logs, while n8n receives this data from Wazuh, and it works as an orchestration tool. It perform as orchestration engine, also providing real-time Gmail alerts, and with the use of Google Cloud Console, in which the OAuth 2.0 credentials are used to secure the connection. This research proves that if two open source tools are combined, they can provide a secure and real-time interface. Also, implementing a severity log system to mitigate alert fatigue. In this case, a logic is applied; if the alert is level 10+, it gives a notification; otherwise, Wazuh can save all information. This system can be easily implemented by a small organization.

The main contributions are as follows:

- Built a cost-effective security system with the help of open source tools like Wazuh and n8n.
- Developed an automated system that can monitor our machine 24/7.
- Using Wazuh as a detection tool to capture logs and send them to n8n (an automation tool) via webhook.
- Integrating the Google Cloud API with OAuth 2.0 token-

DOI: https://doi.org/10.67535/tsp.000003.004
Published under the ICACNC 2026 Proceedings

based authentication to eliminate static password leak and generate Gmail alerts with secure connection.
- Built n8n as a rule-based automation that can only forward alerts to the administrator, which is "Critical" or level 10+ to mitigate alert fatigue.

II. Literature Review

With the rapid increase in cyber threats, it is very challenging to monitor every threat manually. It is essential to detect threats with a SIEM tool, but SOAR tools are the brain that acts as an automation [2], [3]. So these studies focus on survey-based evaluation, but in this, there is a lack of knowledge on the practical hand on testing. This research bridges this gap by deploying the open source and built a working prototype, which is budget-friendly for small organizations. The research shows that with the use of automation tools, an organization with 250 employees can work more efficiently.These studies show that Dockerized infrastructures can provide cost-effective work for SMEs. [4].It faces limitations regarding ,if the organization uses old tools, as this infrastructure cannot handle heavy traffic, along with the need to maintain technical configuration in Docker. Threat monitoring in small organizations is neglected due to the most expensive tools; with the use of Wazuh and n8n, they are cost-effective. Every small organization can maintain its proper system easily.

In many companies, they use an open source tool to monitor their system, but it is also crucial to use a cloud-based tool so that any non-technical person can monitor their device easily [5].This research accepts that it requires a lot of money to deploy Cloud Wazuh. Maintaining an updated firewall remains a challenge. With the use of this, it can prevent a DoS attack, a Metasploit attack, and also prevent FIM if there is a change in the file; it can detect it. [6].This research can focus on a virtual lab environment, which is a very small thing compared to a real-world enterprise. Also, a combination of Wazuh and Osquery can be used to find vulnerabilities in their own system, which is used for deep system monitoring. [7].When Wazuh and Osquery can run at the same time, execution consume system resources (CPU and RAM)it can make systems slow.

Past research demonstrates that with the combination of Wazuh and the automation tool, it can generate a telegram notification on mobile phones [1].The main problem in this research is that Telegram is a cloud-based messaging app. If the internet becomes slow, the security team can not receive critical alerts. It act as a HIDS(Host Intrusion Detection System) [8]. In the research, I used to generate a real-time Gmail notification for professional requirements. On the other hand, researchers can also combine "the Hive" with an instant response tool, and CVSS(Common Vulnerability Scoring System) provides an approach that depend on heavily specialist experts. Also, it can face alert fatigue. [9]

Researchers tell us there is a deep analysis of how to monitor data, detect cyberattacks, and log them in different ways. [10]. Firewalls and antivirus software are not enough for security; an attacker can bypass them through social engineering techniques. SOAR tools are becoming the backbone of cybersecurity [11]. Our research enhances security by utilizing Gmail OAuth 2.0 for Gmail notifications. This is more secure than the old method, because in this method, we used credentials. This OAuth protocol is much more secure than the normal API. In industrial security systems, many organizations use the combination of SIEM and SOAR tools to prevent manual monitoring, and also in the industrial sector, the data is very sensitive in this area [12]. There is a very difficult way to combine IT (Information Technology) and OT (Operational Technology), which is used to generate new security challenges.

One more challenge in SOC is "Alert fatigue". To manage low-severity logs, an automatic orchestration can remove these types of problems because it can separate high-severity logs or low-severity logs [13], as recent research has shown that the SOAR tool is not for a specific domain; it is also used to manage different domains [14]. In the future, the traditional method will be replaced by automation and also it is empowering to focus on high-level cognitive tasks [15]. Research accepts the rule based automation used for defensive systems, but now they are working for an offensive system it means it can generate a security-dangerous environment.

Previous research on cloud applications shows that the API keys are more vulnerable [16], [17], simple API and authentication are not secure because their credential can be stolen easily. On the other hand, OAuth 2.0 is so secure because they used token based credentials.Mostly, research focuses on technical authentication (tokens, biometrics), but social engineering or human error can also be a very big problem in today's world, which can lead to security breaches. It is very difficult for developers to configure and manage the OAuth setup. But it is more secure than other forms of authentication. [18], [19]

In summary, existing techniques researchers focus on alerts and automation tools; however, there are a lot of searches on many tools, but I integrate a unique concept of automation and security, which combines Wazuh log monitoring with Google Cloud OAuth 2.0 and with n8n. This gap motivates us to focus on security.

III. Methodology

A. System Architecture

The proposed system integrates the automated detection system with the combination of two tools,the Wazuh (SIEM) and n8n (SOAR) tools, which can act like an intelligent system.

B. Environment Setup & Configuration

Firstly, the Wazuh agent was installed on the targeted machine, which is used to monitor suspicious activity or collect logs. They can send logs to the Wazuh manager.

We need to install n8n deployed locally using Node.js on localhost:5678, because firstly we need to test it on one machine, n8n connected to a webhook to make a bridge

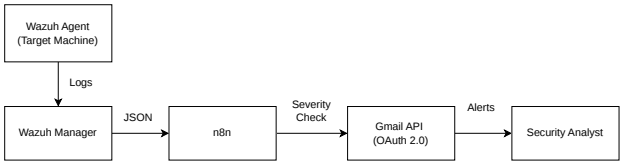

Fig. 1. Proposed System Architecture of Wazuh and n8n Integration

between Wazuh Manager and n8n. After that, the URL of n8n is pasted in the Wazuh manager file for connection to tell the manager where to send data.

C. Log Ingestion & Severity Classification

After that, the Wazuh agent collects logs and sends them to the Wazuh manager, and the Wazuh manager can decode them by using the decoder and rules to generate proper logs or separate the low and high severity logs. This system used to detect different security issues. In some cases, an attacker enters the system using a brute-force attack and generate multiple passwords which can trigger level 10 attack. Wazuh can also detect it and give a notification via Gmail.

Wazuh is a very powerful tool that is not for generating logs. It can give a small detail about the attacker's machine and also the targeted machine. When a high-severity alert is generated, Wazuh sends data in JSON (JavaScript Object Notation)format to the webhook n8n. And then n8n works like a processing engine, it can filter it if the attack severity is greater than or equal to 10, it goes to the Gmail node, else discard it.

D. Secure Integration & OAuth Workflow

When JSON is received by n8n, a Gmail API is used to secure the connection by using the OAuth 2.0 protocol.

In Google Cloud Console, a new project was registered. We use this project to gain access to the Google API, which is used to access the API and services to enable Gmail API in this screen.OAuth was utilized instead of simple API authentication because OAuth is more secure than this. We used to enter the name of the tool that we give access to send mail, and also give the Gmail ID on which they send, and also add the Google scope.

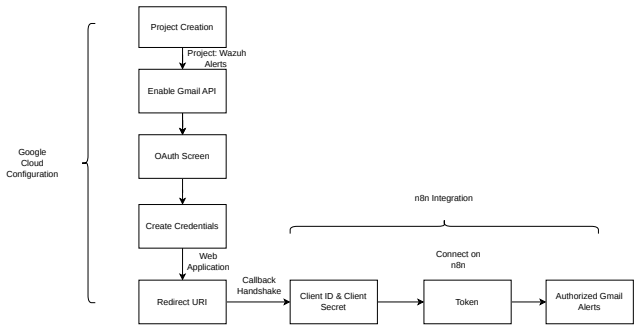

Fig. 2. OAuth 2.0 Setup and Integration Workflow

To enable OAuth 2.0, add credentials and an OAuth client ID. Paste the n8n URL. After that, they give two things: client ID and client secret, which are pasted on the n8n webhook, and click on connect. In this, we use token-based authentication, in this there is no need of sharing original Gmail password by using tokens, and also give n8n just the sent scope; they just sent an email, not read the email.

1) API Configuration: The secure integration is the backbone of this paper, using Google Cloud Console, the details of which are as follows:

a) Google Cloud Console Configuration: A new project was created on Google Cloud because Google used to know which application was using Google services, then used to enable Gmail API in API Services & Library; by default, it is off.

b) Permission Scopes: The OAuth console screen must require permission; it is set on "External." Add scope to it (https://www.googleapis.com/auth/gmail.send), n8n just sent mail in this.It is not allowed to read other emails.

c) Credentials Generation: When the OAuth client ID was created. They gave us secret keys(Client ID & Client Secret) after all procedures. n8n redirected the URI, which was pasted in this project, because Google used to know, after authentication, where to go.

E. Automated Alerting

For testing, we just changed the Wazuh manager settings. They can check the logs which are level>=10 and apply a brute force attack, and also trigger multiple login attempts, and Wazuh detects a login attempt, giving a signal,which is shown on the dashboard and also receives an email. At the end, the webhook published the project, which means to monitor their system 24/7.

IV. Results & Comparison

A. Attack Simulation & Detection

Brute-force simulation was applied to test the system. Multiple failed login attempts on a Windows machine (agent) were collected by Wazuh Manager and considered level 10 attacks under rule ID 60204 (multiple failed login).

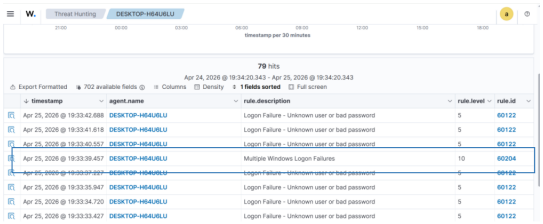

Fig. 3. Wazuh Dashboard showing Level 10 alert

Wazuh is not just for collecting logs; it uses the JSON decoder to get important info like:

- Target user: Victim machine
- Source IP: Attacker machine
- Severity level: Which is set to level 10

This decoder data in JSON format is forwarded to the webhook URL of n8n.

B. Orchestration Process

The n8n platform initiates the workflow upon receiving the JSON payloads and parses the fields. Applying a filter rule. level>= 10, using this, we can reduce "alert fatigue."

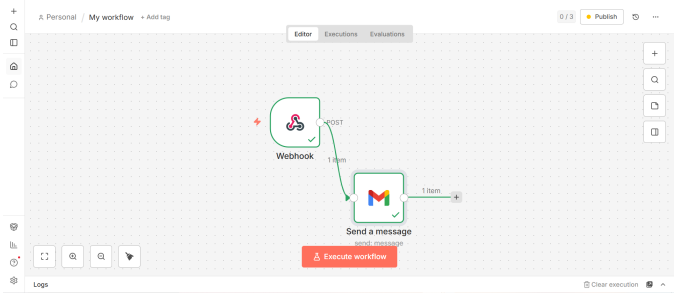

Fig. 4. The n8n workflow execution connected from Webhook node to Gmail node.

C. Secure Notification Output

When the condition is match the criteria, then the Gmail node sends an alert to the administrator. With the use of OAuth 2.0, the process will be secure and make authentication with the use of tokens.

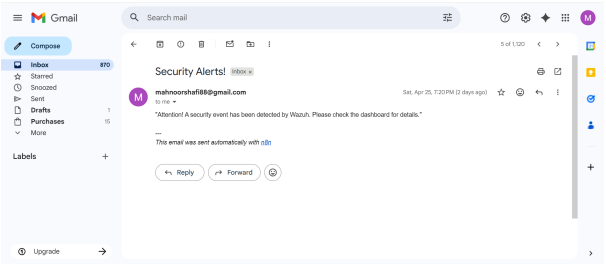

Fig. 5. Automated Gmail notification received through OAuth 2.0.

D. Comparative Analysis

Features	Traditional Method	Proposed Integration
Response	Manual Dashboard Monitoring	Automated Dashboard Monitoring
Data format	Logs/Text file	JSON Format
Security	Used API security	OAuth 2.0 Token Based
Alerts Filter	Send alerts on telegram	Send alerts on Gmail
Alert Fatigue	Generate false positive	If Severity >=10 send alert

E. False Positive & Reliability

The system avoids false positives because Wazuh has smart rules which triggers alerts for real attacks, not normal user activity. Also, n8n safely delivers every true alert to admin. If the internet goes down, Wazuh saves the alerts in its internal memory and sends them automatically when the connection is back.

F. Quantitative Metrics

The proposed system features high efficiency, maintaining an average processing latency between 180 ms and 280 ms for critical notifications because level 10+ alerts logic is applied in this framework. It effectively reduces alert fatigue by compressing 500 repetitive log entries into a single email alert because in n8n deduplication logical node is applied with the use of this it collect same IP address and consider it as one and send detail summary email. Furthermore, the framework operates with a hardware (under 12% CPU and 1.2 GB RAM), making it cost-effective for small organizations.

G. Performance Analysis

This comparison shown in Fig. 6 is based on implemented system features like detection capabilities, workflow automation, security architecture. Metrics are evaluated using a Functional Availability Matrix, where scores are mapped objectively 0= Not exist, 1= Basic, and 2= Advanced, which gives details about the proposed system over the baseline system [1]. According to the feature of complexity and effectiveness, we give these a range from 0 to 2. The existing system is based on static API tokens for Telegram notifications, which are vulnerable; on the other hand, our proposed system is based on OAuth 2.0, which is secure compared to other APIs.In the previous system, there is no concept of security, but our system implements OAuth 2.0 for secure connections.

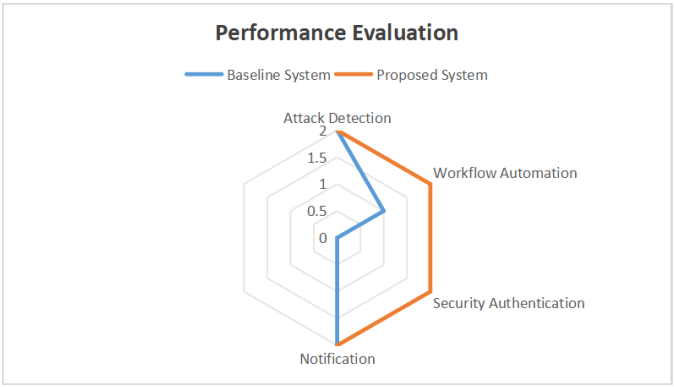

Fig. 6. Proposed System vs Baseline system

H. Security Analysis

- **Authentication:**Traditional alert mechanisms(Telegram or Slack) depend on static API keys. If these keys are leaked the attacker gain unauthorized access.In contrast, our proposed system implements OAuth 2.0, which utilizes short-lived access tokens. Even if a token is leaked, it automatically expires within a short duration, which can make system secure.
- **Principle of Least Privilege:**Within our OAuth 2.0 configuration, the n8n automation workflow is restricted to give a permission scope it is strictly limited to gmail.send. The advantage of this workflow is that even if an attacker compromise the token, they are only authorized

to forward alerts and cannot read, modify, or access the administrator's emails.

- **Limitations:** If an attacker hacks the host machine running n8n, the stored Client Secret could be stolen. To solve this problem, we need to use strict token rotation and secure environment. Moreover, the current setup is deployed on a local machine, which cannot handle huge traffic. For larger environments, this system can be shifted to a Dockerized cluster with load balancing to manage heavy log traffic easily.

V. Conclusion

This research successfully demonstrates the combination of open-source tools that build real-time security alerts in an official Gmail address with a secure connection, operating as incident response workflow. Many companies use expensive tools to secure and maintain their data, whereas we develop a similar system with security features using open-source tools. The experiment shows that Wazuh generates logs even when the severity is low. If the logs are critical, this project gives Gmail alerts. This helps the security team avoid monitoring all threats every time. Using n8n logic to filter non-critical logs and also connecting with Google Cloud Console to enable the Google services to build a secure connection between tools and your personal data. This is not just about connecting nodes; this research is all about the logic and integration. Also, we separate the low-severity and high-severity logs using the Wazuh manager configuration file.

Future work will focus on an "Active Response" to automatically block IP addresses that are repeatedly attacked.

Also used to add custom rules to detect malicious files when they enter the system, or if a malicious USB is inserted into the system, it can generate an alert on Gmail.

References

[1] C. Dhefanni, M. Suhartana, and R. Hassan, "Real-time attack simulation and detection using wazuh and telegram alerts," in *2025 International Conference on Information Management and Technology (ICIMTech)*. IEEE, 2025, pp. 728–733.

[2] S. Dwivedi, B. Rajendran, P. Akshay, A. Acha, P. Ampatt, and S. D. Sudarsan, "Intellisoar: Intelligent alert enrichment using security orchestration automation and response (soar)," in *International Conference on Information Systems Security*. Springer, 2024, pp. 453–462.

[3] S. Norem, A. E. Rice, S. Erwin, R. A. Bridges, S. Oesch, and B. Weber, "A mathematical framework for evaluation of soar tools with limited survey data," in *European Symposium on Research in Computer Security*. Springer, 2021, pp. 557–575.

[4] L. F. Ilca, O. P. Lucian, and T. C. Balan, "Enhancing cyber-resilience for small and medium-sized organizations with prescriptive malware analysis, detection and response," *Sensors*, vol. 23, no. 15, p. 6757, 2023.

[5] S. Moiz, A. Majid, A. Basit, M. Ebrahim, A. A. Abro, and M. Naeem, "Security and threat detection through cloud-based wazuh deployment," in *2024 IEEE 1st Karachi Section Humanitarian Technology Conference (KHI-HTC)*. IEEE, 2024, pp. 1–5.

[6] H. Javid, "Practical applications of wazuh in on-premises environments," 2024.

[7] M. D. Prasad, M. S. N. S. Sindusha, N. Jahnavi, M. W. Ali, and S. A. Devi, "Enabling cybersecurity defenses: Advanced endpoint detection, data breach identification, and anomaly resolution," in *2024 8th International Conference on Inventive Systems and Control (ICISC)*. IEEE, 2024, pp. 461–468.

[8] S. Stanković, S. Gajin, and R. Petrović, "A review of wazuh tool capabilities for detecting attacks based on log analysis," *No Nama Agent Integrity File Added Delete Modified*, vol. 1, 2022.

[9] B. Soewito *et al.*, "Siem and threat intelligence: Protecting applications with wazuh and thehive." *International Journal of Advanced Computer Science & Applications*, vol. 15, no. 9, 2024.

[10] R. Iudica, "A monitoring system for embedded devices widely distributed," Ph.D. dissertation, Politecnico di Torino, 2022.

[11] A. Sridharan and V. Kanchana, "Siem integration with soar," in *2022 International Conference on Futuristic Technologies (INCOFT)*. IEEE, 2022, pp. 1–6.

[12] S. Chatterjee, "Using siem and soar for real-time cybersecurity operations in oil and gas," *International Journal of Innovative Research and Creative Technology*, vol. 6, no. 2, pp. 1–11, 2020.

[13] R. A. Bridges, A. E. Rice, S. Oesch, J. A. Nichols, C. Watson, K. Spakes, S. Norem, M. Huettel, B. Jewell, B. Weber *et al.*, "Testing soar tools in use," *Computers & Security*, vol. 129, p. 103201, 2023.

[14] E. A. Hadi, E. Rusnandi *et al.*, "Automation of technical documentation validation workflows based on github using n8n during software development stages," *Journal of Multimedia Technology and Applied Software*, vol. 2, no. 2, pp. 64–69, 2025.

[15] S. J. Lazer, K. Aryal, M. Gupta, and E. Bertino, "A survey of agentic ai and cybersecurity: Challenges, opportunities and use-case prototypes," *arXiv preprint arXiv:2601.05293*, 2026.

[16] F. Qazi, "Application programming interface (api) security in cloud applications," *EAI Endorsed Transactions on Cloud Systems*, vol. 7, no. 23, p. e1, 2023.

[17] F. Hussain, B. Noye, and S. Sharieh, "Current state of api security and machine learning," *IEEE Technology Policy and Ethics*, vol. 4, no. 2, pp. 1–5, 2022.

[18] A. Gupta, M. Panda, and A. Gupta, "Advancing api security: a comprehensive evaluation of authentication mechanisms and their implications for cybersecurity," *International Journal of Global Innovations and Solutions (IJGIS)*, 2024.

[19] S. Matcha and K. Munish, "Enhancing software security with oauth 2.0: Implementation strategies and vulnerability mitigation," *Enhancing Software Security with OAuth*, vol. 2, no. 12, p. 3, 2025.