I. INTRODUCTION

SQL injection attacks pose a significant threat to web applications and database systems. These vulnerabilities are frequently exploited by attackers to compromise website security and gain unauthorized access to sensitive information [1]. SQL injection attacks involve the injection of malicious code into a database-driven web application, allowing attackers to manipulate the database and extract or modify data [2]. As interactive web applications that rely on backend database services proliferate, the prevalence of SQL injection attacks increases [3].

The consequences of SQL injection attacks are severe, leading to unauthorized access to databases, information leakage, and data falsification in web applications [4]. These attacks not only result in data breaches but also compromise data integrity, disrupt server operations, and damage organizational reputation [5]. Consequently, preventing SQL injection attacks is crucial to maintaining the confidentiality and integrity of data stored in databases [6].

To address this threat, researchers have explored various approaches, including prevention techniques such as complementary character coding and enhanced parameterized stored procedures, which effectively reduce the risk of injecting malicious code. For instance, Ahmad and Karim [5] developed an advanced parameterized stored procedure method, significantly enhancing security by preventing SQL injection attacks. Similarly, Mui and Frankl [6] proposed combining complementary character coding to prevent web application injections, effectively mitigating SQL injection vulnerabilities.

Integrating SIEM systems with multi-Wazuh agents and heterogeneous WAFs presents a powerful approach for detecting threats collaboratively SQL injection attacks. The multi-Wazuh agents efficiently monitor and collect security data from various endpoints, which the SIEM system can then centrally analyze to identify malicious SQL injection attempts[7]. WAFs also play a crucial role in protecting web applications from injection attacks, including SQL injection[8]. By combining these technologies, organizations can establish a proactive security posture that not only detects SQL injection attacks but also responds to them in real time, thereby reducing the potential risk to their web applications and databases

This study presents a novel approach by integrating the Wazuh SIEM platform with multiple Web Application Firewalls (WAFs), specifically ModSecurity and NAXSI, for collaborative detection and mitigation of SQL injection attacks. The key contributions of this research are outlined as follows:

- **Integration of diverse WAFs and SIEM:** This study integrates Wazuh SIEM with ModSecurity and NAXSI to enhance the detection of SQL injection threats, focusing on Time-Based, Error-Based, and Union-Based SQL injection techniques.
- **Performance analysis of WAFs:** The study compares the effectiveness of ModSecurity and NAXSI in detecting SQL injection attacks. ModSecurity outperformed NAXSI in Time-Based and Error-Based attacks, while both WAFs showed equal effectiveness in mitigating Union-Based attacks.
- **Centralized threat monitoring:** Using the Wazuh platform, the system efficiently collects and visualizes log data in real-time, improving threat detection and response times for security teams

This research provides a scalable and efficient framework for enhancing web application security against SQL injection attacks by leveraging the strengths of multiple WAFs integrated within a SIEM platform