|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Paper title | Authors | Problem | Methodology | Result | Draw back |
| Pixy: A Static Analysis Tool for Detecting Web Application Vulnerabilities | Nenad Jovanovic, Christopher Kruegel, and Engin Kirda | The problem addressed in the paper is the increasing number and importance of web applications coupled with the growing quantity and impact of security vulnerabilities within them. | * static source code analysis * literal analysis and alias analysis | The static analysis employed by the Pixy application demonstrates its ability to detect security vulnerabilities effectively in real-world web applications. | * They addressed SQL injection vulnerabilities but operated on a less ambitious conceptual level compared to other approaches. * didn't perform alias analysis |
| WebGuardia – An Integrated Penetration Testing System to  Detect Web Application Vulnerabilities | Nisal Madhushan Vithanage,  Neera Jeyamohan | Due to the spread of web applications and the exchange of sensitive parameters through them, the risks of cyber attacks targeting vulnerabilities through web applications have increased. Because of the problems resulting from traditional testing tools, such as generating high rates of false positives or negatives, inefficiency, and lack of reliability in the results. It has become necessary to have a more effective tool to discover the most common. | The paper presented WebGuardia, an integrated intrusion system that aims to identify five of the top ten vulnerabilities found in web applications identified by OWASP. The system focuses on SQLI and XSS detection. WebGuardia uses a simple architecture that includes crawling, attack, analysis, and reporting modules. It is considered an advanced update to existing open source detection techniques. | Based on the results of the evaluation of WebGuardia that was conducted on many web applications and more than 700 web pages, which appeared in a comparison table between them and the three most famous open source tools for discovering web vulnerabilities: OWASP ZAP, w3af, and VEGA. It showed that WebGuardia is able to detect vulnerabilities more accurately compared to other tools. | The paper focused on some web vulnerabilities and ignored many of them, which may lead to providing inaccurate estimates of security vulnerabilities. (A comprehensive analysis of security vulnerabilities was not provided). |
| Paper title | Authors | Problem | Methodology | Result | Draw back |
| A STUDY ON SQL INJECTION TECHNIQUES | Rubidha Devi, Ramasamy Venkatesan, Raghuraman Koteeswaran | SQL Injection vulnerability | collected and analyzed for the purpose of understanding SQL injection techniques and verifying how they work and their impact on database security. | Implementing robust input validation and parameterized queries can mitigate the risk of SQL injection attacks, bolstering database security effectively. | The complexity and maintenance overhead of implementing robust input validation and parameterized queries may pose challenges, and they might not fully eliminate the risk of SQL injection. |
| Deep Dive into Directory Traversal and File Inclusion Attacks leads to Privilege Escalation | Mrunalsinh Chawda, Dr. Priyanka Sharma, Mr. Jatin Patel | Directory Traversal and File Inclusion Attacks(Vulnerability) | Theoretical research on different penetration techniques, analyzing potential threats, reviewing security vulnerabilities in software and systems, and testing those vulnerabilities in experimental or real-world environments. | 1. Enforce rigorous input validation to sanitize user inputs and prevent malicious traversal attempts. 2. Implement strict access controls to restrict file access and minimize the impact of successful attacks. 3. Regularly update software, conduct security audits, and educate personnel to maintain robust defenses against directory traversal vulnerabilities.   Top of Form | Implementing measures against directory traversal vulnerabilities may hinder user experience with overly stringent input validation and add complexity to system management through strict access controls. Additionally, the need for regular updates and security audits can disrupt workflow, impacting productivity. Balancing security with usability is crucial to effectively manage these drawbacks. |