ABSTRACT

. SQL injection remains one of the most critical threats to modern web applications. One variant, Time-Based Blind SQL Injection (TBB-SQLi), is even more dangerous because it does not produce direct output and relies solely on delay intervals as an indicator of successful injection. The main challenge with this technique is that the exploitation process is slow and requires a large number of requests. This research developed a Python-based TBB-SQLi exploitation automation system by implementing three search algorithms: Linear Search, Binary Search, and Ternary Search to determine the efficiency of exploitation in the process of extracting sensitive data from the database. Performance evaluation was carried out using two main parameters, namely execution time (runtime) and number of requests, in six sleep delay scenarios (1–6 seconds). The test results show different performance patterns in each delay scenario, indicating that Binary Search is the most efficient algorithm, with a runtime of 2,054–11,007 seconds and 1,945–2,113 requests. Ternary Search ranked in the middle with a runtime of 3,720–10,361 seconds and 2,652–4,552 requests, while Linear Search was the slowest at low delays, with a runtime of 3,780–6,932 seconds and the highest total number of requests reaching 6,167. The resulting system is capable of automatically extracting databases, tables, and administrator credentials. Overall, Binary Search proved to be the most optimal in balancing time efficiency and the number of requests in the exploitation process.

Keywords : *Time-Based Blind SQL Injection, Exploit Automation, Binary Search, Linear Search, Ternary Search, Web Security*