A Hot Query Bank approach to improve detection performance against SQL injection attacks

Yu-Chi Chung\textsuperscript{a,*}, Ming-Chuan Wu\textsuperscript{b}, Yih-Chang Chen\textsuperscript{b}, Wen-Kui Chang\textsuperscript{b}

\textsuperscript{a}Department of Computer Science and Information Engineering, Chang Jung Christian University, 396 Chang Jung Rd., Sec.1, Kway Jen, Tainan 71101, Taiwan
\textsuperscript{b}Department of Information Management, Chang Jung Christian University, Tainan, Taiwan

\textbf{abstract}

SQL injection attacks (SQLIAs) exploit web sites by altering backend SQL statements through manipulating application input. With the growing popularity of web applications, such attacks have become a serious security threat to users and systems as well. Existing dynamic SQLIA detectors provide high detection accuracy yet may have ignored another focus: efficiency. Our research has found that inside most systems exist many hot queries that current SQLIA detectors have repeatedly verified. Such repetition causes unnecessary waste of system resources.

The research has completed Hot Query Bank (HQB), a pilot design that can cooperate with the existing SQLIA detectors in web applications and enhance overall system performance. HQB simply records hot queries and skip the detector’s verification process on their next appearances. Algorithms for the design have been proposed. A series of simulated experiments has been conducted to observe the performance improved from the design with three respective detectors, SQLGuard, SQLrand, and PHPCheck.

The results have illustrated that utilization of HQB can indeed improve system performance by 45\% of execution time, regardless of different detectors being tested. With such improvement and robustness, the result promises to provide an add-on feature for SQLIA detectors in protecting web applications more efficiently. Future works include further validation of the design in a real web application environment, development of a standard interface to collaborate with web applications and detectors, etc.

\textsuperscript{ª}2011 Elsevier Ltd. All rights reserved.

1. Introduction

SQL injection is a type of security vulnerability in the database layer of a web application (Halfond et al., 2006; Halfond and Orso, 2005a; Mitropoulos and Spinellis, 2009). SQL injection attacks (SQLIAs) exploit web sites by altering backend SQL statements through manipulating application input. With the growing popularity of Web applications, such attacks have become a serious security threat to users and systems as well.

These attacks occur especially when the SQL statements are combined with hard-coded strings for user inputs to create dynamic queries. If a user input is not properly validated, attackers may be able to change the developer’s intended SQL command by inserting new SQL keywords or operators through specially crafted input strings. SQLIAs leverage a wide range of mechanisms and input channels to inject malicious commands into a vulnerable application (Alon et al., 1996).

The following example illustrates how an attacker can...