Chapter 26
A Learning-based Neural Network Model for the Detection and Classification of SQL Injection Attacks

Naghmeh Moradpoor Sheykhkanloo
Edinburgh Napier University, UK

ABSTRACT

Structured Query Language injection (SQLi) attack is a code injection technique where hackers inject SQL commands into a database via a vulnerable web application. Injected SQL commands can modify the back-end SQL database and thus compromise the security of a web application. In the previous publications, the author has proposed a Neural Network (NN)-based model for detections and classifications of the SQLi attacks. The proposed model was built from three elements: 1) a Uniform Resource Locator (URL) generator, 2) a URL classifier, and 3) a NN model. The proposed model was successful to: 1) detect each generated URL as either a benign URL or a malicious, and 2) identify the type of SQLi attack for each malicious URL. The published results proved the effectiveness of the proposal. In this paper, the author re-evaluates the performance of the proposal through two scenarios using controversial data sets. The results of the experiments are presented in order to demonstrate the effectiveness of the proposed model in terms of accuracy, true-positive rate as well as false-positive rate.

INTRODUCTION

SQL is a programming language designed for handling data in a Relational Database Management System (RDBMS) (SQL Introduction, 2016). SQLi attack is a technology weakness that comes from dynamic script language such as PHP: Hypertext Processor (PHP), Active Server Pages (ASP), Java Server pages (JSP) and Common Gateway Interface (CGI). It takes advantages of inappropriate and/or poor coding of web applications that allows hackers to inject malformed SQL commands in order to gain unauthorised access to data resides in the related back-end database.

DOI: 10.4018/978-1-7998-0414-7.ch026

For any organisation, data contains important and confidential information that can be related to them, their customers, and their business partners. This information can range from personal or less sensitive information such as: first name and last name to more sensitive information such as: username, password, pin code, and credit card information. If inputs from a user-side are not properly sanitised, a hacker can generate crafted SQL commands and can inject them into a database in order to pass say a login barrier and see what exists behind it. This leads to sever damages on a given database such as: disclosing, modifying, and/or removing data or in a worse-case scenario wiping the entire database. Therefore, it is important for any organisation to protect their databases in order to prevent any loss to themselves, their customers, and their business partners.

SQLi attack has been ranked as the most harmful danger, A1-Injections, in top 10 security threats for web applications in Open Web Application Security Project (OWASP) (Top 10, 2013). An A1-Injection attack includes injection flaws such as: SQL, OS, and LDAP injections. This occurs when unsafe and/or untrusted data is sent to an interpreter as part of a command or query tricking it into executing unintended commands or accessing data without a proper authorisation.

CIA triad, which stands for: Confidentiality, Integrity, and Availability, is a well-known security model that can be used to develop a security policy for any organisation. If a given database is attacked, CIA elements can be violated. For instance, the data in the database can be revealed to unauthorised users, which is a failure in Confidentiality element of the CIA triad. The data can be altered, which is a failure in Integrity element of the CIA triad. In a worst-case scenario, the data can be completely wiped out from the database which is a failure in Availability element of the CIA triad.

In the previous work (Moradpoor, 2014), the author proposed a NN-based model for SQLi attack detections which built from three elements: a URL generator, a URL classifier, and a NN model. Addressing the published results, the previous proposal was successful to detect the malicious URLs from the benign URLs. The author then extended the proposal to a pattern recognition NN-based model for the detection and classification of the SQLi attacks (Moradpoor, 2015). Addressing the published results, the proposed model was successful to not only detect the malicious URLs from the benign URLs, but also classify the malicious URLs into the popular SQLi attack categories. Finally, in the most recent work (Moradpoor, 2015, SQL-IDS), the author stress tested the previous proposals where the model demonstrated a good performance in terms of accuracy. In this paper, the author further investigates the performance of the previous proposal (Moradpoor, 2014; Moradpoor, 2015; Moradpoor, 2015, SQL-IDS) by implementing two different test beds and scenarios. This includes employing different sets of data for the developed NN-based model in order to demonstrate the effectiveness of the proposed technique.

The remainder of this paper is organised as follows. In Sections II, the author reviews the related work for the detections and preventions of the SQLi attacks. The author's previous proposal (Moradpoor, 2014; Moradpoor, 2015; Moradpoor, 2015, SQL-IDS) and the related implementations are discussed in Sections III & IV, respectively. Sections V and VI include two different scenarios along with the related results using three sets of data. This is followed by conclusions of the work in Section VII, acknowledgments, and references.