# A Multiagent Based Strategy for Detecting Attacks in Databases in a Distributed Mode

Cristian Pinzón[1], Yanira De Paz[2], and Javier Bajo[1]

[1] Departamento Informática y Automática, Universidad de Salamanca, Plaza de la Merced s/n 37008, Salamanca, Spain
[2] Universidad Europea de Madrid, Tajo s/n 28670, Villaviciosa de Odón, Spain
cristian_ivanp@usal.es, yanira@usal.es, jbajope@usal.es

**Abstract.** This paper presents a distributed hierarchical multiagent architecture for detecting SQL injection attacks against databases. It uses a novel strategy, which is supported by a Case-Based Reasoning mechanism, which provides to the classifier agents with a great capacity of learning and adaptation to face this type of attack. The architecture combines strategies of intrusion detection systems such as misuse detection and anomaly detection. It has been tested and the results are presented in this paper.

**Keywords:** Multi-agent, SQL injection, Security database, case-based reasoning, IDS.

## 1 Introduction

The exponential growth of the computer network and the increase in the interconnection between networks has extended the offer of new services within the cyberspace [1]. The information volume with a sensitive value for the organizations is stored on information structures denominated databases and this information generally is transmitted across computer network. Databases are the core of many information systems, reason for which databases are increasingly coming under large number of attacks. Every day are founded new vulnerabilities in security systems intended to protect databases. These vulnerabilities are used by hackers in order to carry out attacks on the stored data. A special intrusion type within of databases is the SQL injection attack, which occurs when the intended effect of a SQL sentence is changed by inserting SQL keywords or special symbols [2].

Nowadays, the majority of approaches had addressed the problem of SQL injection attack from a centralized perspective, such as the one described by [3] and [2]. However, the solutions are limited to solve only a part of the problem. Regarding this, other approaches had implemented strategies based on intrusion detection systems in order to block a SQL injection attack, such as [4] and [5]. These proposals have as main drawbacks the highest error rate and a limited capacity of learning and adapting when changes occur in the patterns of attacks.

Our proposal aims the SQL injection attacks in a distributed, dynamic and flexible mode. This proposal is founded in a hierarchical multiagent architecture using agents based on the BDI (Belief, Desire and Intention) model [6]. Agents are typically integrated into multiagent systems or agent societies, exchanging information and resolving problems in a distributed way [7]. Agents can be characterized through their capacities such as autonomy, reactivity, pro-activity, social abilities, reasoning, learning and

J.M. Corchado et al. (Eds.): DCAI 2008, ASC 50, pp. 180–188, 2009.
springerlink.com 

mobility [6]. Our proposal incorporates classifier agents supported by a Case-based reasoning mechanism (CBR) [8] that includes a mixture of neural networks capable of making short term predictions [9]. Our multi-agent architecture is adequate to block the SQL injection attack, because it is designed for working in distributed and dynamic environments.

The rest of the paper is structured as follows: section 2 presents the problem that has prompted most of this research work. Section 3 focuses on the details of the multiagent architecture, the different levels of the architecture, the interaction possibilities and communication between the agents; section 4 explains in detail the classification model integrated within the classifier agent. Finally, section 5 describes how the classifier agent has been tested inside a multi-agent system and presents the results obtained.

## 2 SQL Injection Attacks Description

The impact of a SQL injection attack in a database has many consequences within of the organization and individuals. Personal, financial and legal information is compromised when this type attack is carried out. A SQL injection attack takes place when a hacker changes the semantic or syntactic logic of a SQL text string by inserting SQL keywords or special symbols on the original SQL command that will be executed at the database layer of an application [10], [2]. The results of this attack can produce unauthorized handling of data, retrieval of confidential information, and in the worst possible case, taking over control of application server. One particular inconvenient of the SQL injection attack is the biggest number of variants. Some strategies can be extremely complex due to the high number of variables that they can generate, thus making their detection very difficult.

Some approaches based on firewall and intrusion detection system (IDS) are a few effective due the strategy of detection, which requires an updated patterns database. Other approaches more specifics to face SQL injection attacks are founded in a technique of string analysis, some carrying out static analysis such as JSA (Java String Analyzer) [3]. Other more complex using dynamic and hybrid analysis is AMNESIA (Analysis and Monitoring for Neutralizing SQL Injection Attacks) [2]. These approaches generally have as main drawback that aim just one part of the problem, moreover the approaches based on models for detecting SQL injection attacks are very sensitive. With only slight variations of accuracy, they generate a large number of false positive and negatives.

Several approaches based on artificial techniques and hybrid systems propose a novel alternative. Web Application Vulnerability and Error Scanner (WAVES) [11] uses a black-box technique which includes a machine learning approach. Valeur [4] presents an IDS approach which uses a machine learning technique based on a dataset of legal transactions. These transactions are used during the training phase prior to monitoring and classifying malicious accesses. Rietta [5] put forward an IDS at the application layer using an anomaly detection model. Finally, Skaruz [12] proposes the use of a recurrent neural network (RNN). The detection problem is became a time serial prediction problem. Generally, this approaches present as main problem, generating a large number of false positive and false negative. In the case of the IDSs systems, they are unable to recognize unknown attacks because they depend on a signature database that requires a dynamic updating.