MIECT: Security 2020-21

Practical Exercises: SQL injection attack

January 12, 2021 Due date: no date

Changelog

- v1.0 - Initial Version.

1 Introduction

SQL injection attacks represent a serious threat to any database-driven site. The methods behind an attack are easy to learn and the damage caused can range from considerable to complete system compromise. Despite these risks, an incredible number of systems on the Internet are susceptible to this form of attack.

Not only is it a threat easily instigated, it is also a threat that, with a little common-sense and forethought, can easily be prevented.

It is always good practice to clean and validate all input data, especially data that will used in OS command, scripts, and database queries, even if the threat of SQL injection has been prevented in some other manner.

For complete security bind your parameters to the SQL query and do not create string based SQL queries with user (or other external) data.

2 SQL injection attacks

This guide will be based on the WebGoat application, which can be downloaded from `https://github.com/WebGoat/WebGoat/releases`; get the latest version (currently v8.1.0) of the server JAR file.

After WebGoat is installed, run it with the `java` command (you need to use a JVM with version 11 or above). Access the WebGoat server (`http://localhost:8080/WebGoat`) with a browser, create an account and proceed to the set of attacks denominated "(A1) Injection", which focus on SQL injection issues.

It will be useful to install the "Tamper Data" Firefox extension. This will allow for easily intercept HTTP POST requests, and modify them in order to add injection statements into POST parameters.

Follow the SQL injection introduction topics, which contains several explanations and experiments to get aquainted with SQL injection issues.