INFORMATION SECURITY - CS3002

Project Proposal: Prevention of SQL injection attacks using regular expressions

Onais Ali Shah 21K – 4691

Mohammad Zohaib 21K –3215

Muhammad 21K– 3192

Project Objectives

The primary objective of this project is to develop a lightweight program capable of analyzing user input to determine if it is malicious and poses a threat to an application's database. For this, we use Python socket programming to build an efficient system for real-time detection of SQL injection attacks. Objectives of this project include:

- **Enhanced Security**: Design a system for detecting and mitigating SQL injection attempts by analyzing user inputs in real-time.
- **Efficiency**: Implement a lightweight and resource-friendly solution suitable for environments with limited computational capacity.
- **Scalability**: Ensure the solution can handle multiple simultaneous connections without compromising performance.
- **Practical Implementation**: Test the system against various attack scenarios to evaluate its effectiveness and reliability.

Proposed Solution

The proposed solution involves the use of Python socket programming to create a client-server model. The server will analyze incoming client inputs, detect malicious patterns, and take appropriate action, such as disconnecting the client if a SQL injection attempt is identified.

Key Features:

- **Regular Expression Matching**: The server will employ predefined patterns to identify SQL injection attempts, such as `UNION SELECT`, `DROP TABLE`, and `OR 1=1`.
- **Real-Time Analysis**: The system will process and evaluate each user input in real-time, ensuring timely action against potential threats.
- **Automatic Disconnection**: If a malicious input is detected, the server will terminate the connection to prevent further harm.
- **Logging**: The system will log all inputs that are registered as harmful and safe separately, allowing for later inspection.

Implementation Details

The implementation will consist of the following components:

- **Server Module**:
  - A Python-based server capable of handling multiple simultaneous client connections using sockets.
  - Scans incoming inputs and applies regular expression-based filters to detect malicious patterns.
- **Client Module**:
  - A lightweight client program that connects to the server and allows users to send input strings.
  - Simulates user interaction and potential attack scenarios for testing purposes.

- **Detection Logic**:
  - Regular expressions designed to identify SQL injection patterns, including WHERE, UPDATE, and multi-line statements.
  - Adaptable logic to enhance detection accuracy and prevent false positives.
- **Logging and Action**:
  - Logs all detected malicious inputs for analysis and improvement of detection rules.
  - Disconnects clients sending malicious inputs while allowing legitimate users to continue operations.

Conclusion

This project addresses a critical aspect of application security by creating a lightweight yet effective system for detecting SQL injection attacks. By leveraging Python's socket programming and regular expressions, the proposed solution ensures enhanced security without the computational overhead associated with traditional WAFs.

The outcomes of this project could contribute to improving security practices for small-scale applications, providing an accessible solution for environments with limited resources. Furthermore, the findings may inspire future research into lightweight, real-time security solutions capable of handling evolving cyber threats.