DETECTING DATA LEAKS USING SQL INJECTION

Submitted by

Name: Varsha

College: Kathir College of Engineering

Department: Computer and communication Engineering

Year: 2025–2026

1. Introduction

SQL Injection is one of the most serious security vulnerabilities in cloud-based applications. It occurs when attackers exploit improperly validated user inputs to manipulate SQL queries and gain unauthorized access to sensitive data. In cloud systems, SQL injection attacks can lead to massive data leaks, financial losses, and compromise of user privacy. This project focuses on building a secure cloud system that detects and prevents data leaks caused by SQL injection attacks.

2. Problem Statement

Cloud applications store large volumes of confidential user information such as login credentials, personal data, and financial records. Many systems fail to properly validate inputs, making them vulnerable to SQL injection attacks. These attacks allow unauthorized users to access or modify sensitive data. Therefore, a secure system is required to protect cloud databases from SQL injection and prevent data leakage.

3. Objectives

To secure user data against SQL injection attacks

To encrypt sensitive data using AES-256 encryption

To implement a capability code mechanism for secure SQL execution

To provide a double-layer security protocol

To make the system lightweight and accessible over the internet

4. System Architecture

The system architecture consists of the following components:

User Interface

Input Validation Layer

Capability Code Generator

AES-256 Encryption Module

Secure SQL Engine

Cloud Database

Monitoring and Logging Module

Each module works together to ensure secure data handling and prevent SQL injection-based data leaks.

5. Module Description

The User Interface allows users to access the system online. The Input Validation Layer sanitizes user inputs to block malicious SQL commands. The Capability Code Mechanism controls which SQL operations are permitted. The AES-256 Encryption Module encrypts sensitive data before storage. The Secure SQL Engine executes only authorized queries. The Monitoring Module detects and logs suspicious activities.

6. AES-256 Encryption

AES-256 is a highly secure symmetric encryption algorithm used to protect sensitive information. In this system, user credentials and confidential data are encrypted before being stored in the cloud database. Even if attackers gain access to the database, the encrypted data remains unreadable.

7. Capability Code Mechanism

Capability codes are secure tokens that define what actions a user can perform on the database. SQL queries are executed only when a valid capability code is provided. This prevents unauthorized access and restricts SQL execution.

8. Double-Layer Security Protocol

The first layer focuses on input validation and SQL injection detection. The second layer enforces encryption and capability-based access control. Together, these layers provide strong protection against data leaks.

9. Algorithm

Accept user input

Validate and sanitize input

Verify capability code

Encrypt sensitive data using AES-256

Execute secure SQL query

Monitor and log activity

10. Advantages

Prevents SQL injection attacks

Protects sensitive user data

Ensures secure cloud database access

Lightweight and efficient system

Easy internet accessibility

11. Applications

Cloud-based web applications

Online banking systems

E-commerce platforms

Healthcare information systems

Educational portals

12. Conclusion

The Detecting Data Leaks Using SQL Injection system provides a secure solution for protecting cloud databases from SQL injection attacks. By combining AES-256 encryption, capability code mechanisms, and a double-layer security model, the system ensures confidentiality, integrity, and reliability of user data