# Detecting Data Leaks via SQL Injection Prevention on an E-Commerce

Karan Ray, Nitish Pol, Suraj Singh Guided by Prof. SUVARNA ARANJO

**Abstract**— In this paper an attempt has been made to develop an online shop that allows users to check for different cloths for women's available at the online store and can purchase cloths online. The project consists of list of cloths displayed in various materials and designs. The user may browse through these products as per categories. If the user likes a product, he/she can add it to his/her shopping cart. Once user wishes to checkout he must register on the site first. Once the user makes a successful transaction admin will get report of his bought products. The objective of this project is to develop a secure path for transaction done by the user. Using AES (Advanced Encryption Standard) encryption technique, the transaction and user account details can be made secured. AES encryption is also used to encrypt the user's card and password information while transaction.

**Index Terms**— Database Security; SQL injection technique; Attack Detection; Attack Preventation ; AES Algorithm;Encryption.

## 1 INTRODUCTION

THIS project is to develop a secure path for transaction done by the user. Using AES (Advanced Encryption Standard) encryption technique, the transaction and user account details can be made secured[3].

An online shop that allows users to check for different cloths for women's available at the online store and can purchase cloths online. The project consists of list of cloths displayed in various materials and designs. The user may browse through these products as per categories. If the user likes a product, he/she can add it to his/her shopping cart. Once user wishes to checkout he must register on the site first[2]. He can then login using same id password next time. Now user may pay through a Card. Once the user makes a successful transaction admin will get report of his bought products. Here we use notepad++ to make the entire frontend. The middle tier or code behind model is designed in PHP and SQL Serves as a backend to store product data thus the online shopping project brings an entire clothing shop online and makes it easy for both buyer and seller to make deals. Admin can add data about their subscribers and it will be viewed by user.

## 2 PROBLEM STATEMENT

This project is to prevent SQL injection while firing queries to database and to make the database secured. This system is online so no need of implementation. It can be accessed through internet from anywhere. The system uses SQL Injection mechanism to keep the data safe and secured[1].

The highlighted part here is encryption of card data using AES (Advanced Encryption Standard) technique. The Online Shop secures the card payment and won't let the card data to get hacked. While user doing a card payment, all the card data is encrypted and then stored into database. System also keeps user details in an encryption form using AES encryption. The system is built of handling SQL Injection capabilities which doubles the security of database and prevents from injection hacking codes into the database. Here, the project files and a database file will be stored into the cloud which will form a connection between application and cloud server via internet[4]. The project will be accessed in the web browser through Azure link..

## 3 Relevance of the Project

.

- This system can be used in single clothing shops.
- This system can be used to sell like chain of clothing shops from a single site.
- Secured transaction while doing card payment.
- Less risk of data getting hacked.
- Use of Standardized AES algorithm for data security.
- The system is very secure and robust in nature.
- SQL Injection prevention mechanism is used

## 4 OBJECTIVE &SCOPE

The objective of this project is to develop a secure path for transaction done by the user. Using AES (Advanced Encryption Standard) encryption technique, the transaction and user account details can be made secured.

This system is online so no need of implementation. It can be accessed through internet from anywhere. The system uses AES encryption to encrypt the user's card and password information while transaction.

## 5 LITERATURE REVIEW

After reviewing numerous electronic journals, articles from IEEE/FCI journals and gathered information provides sufficient knowledge about SQL injection, it's attacking methodology and its prevention. The successive review of distinctive papers is presented herewith.

Gregory T. Buehrer, Bruce W. Weide, and Paolo A. G. Sivilotti, in the research paper “Using Parse Tree Validation to Prevent SQL Injection Attacks” publishes there commendable work in

2005. The techniques for sql injection discovery seems impressive as this paper also covered the SQL parse tree validation very specifically which is mentionable[1].

Zhendong Su and Gary Wassermann, in 2006 published their research in SQL injection entitled “The Essence of Command Injection Attacks in Web Applications” and covered the specific methods to check and sanitize input query using SQLCHECK, it use the augmented questions and SQLCHECK grammar to validate query[2].

Stephen Thomas and Laurie Williams, worked so long in the field of SQL injection and in 2007, in their research paper, entitled “Automated Protection of PHP Applications against SQLinjection Attacks” they covered an authentic scheme to protect application automatically from SQL injection intrusion. This authentic approach combines static, dynamic analysis and intelligent code re-engineering to secure existing properties[3].

Ke Wei, M. Muthuprasanna & Suraj Kothari in 2007, published their work entitled “Preventing SQL Injection Attacks in Stored Procedures” and through this they provides a novel approach to shield the stored procedures from attack and detect SQL injection from websites. Their method includes runtime check with subsequent application code analysis to eliminate vulnerability to attack. The key method behind this vulnerability attack is that it modifies the composition of the original SQL statement and identifies the SQL injection attack.

## 6 SYSTEM DESIGN

### 6.1 System Block Diagram

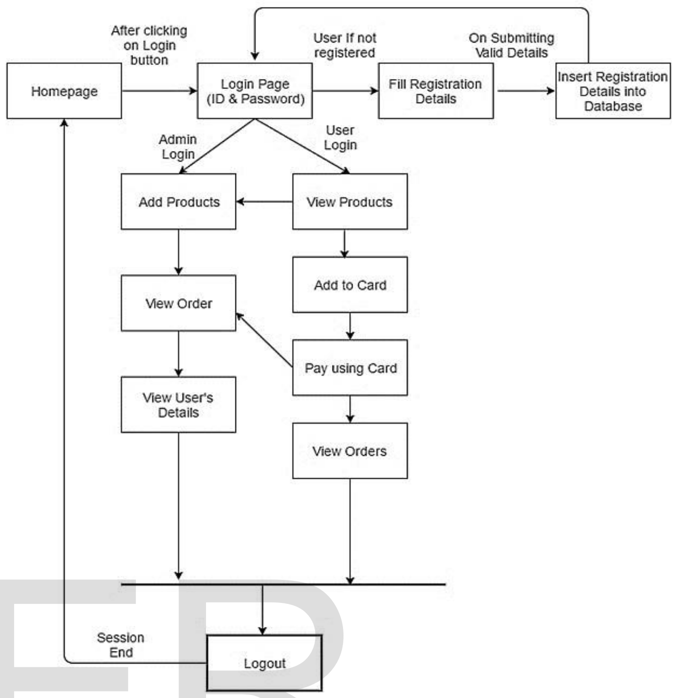


**Fig. 6.1: System Block Diagram**

This is a diagram of a system in which the principal parts or functions are represented by blocks connected by lines that show the relationships of the blocks.

### 6.2 Use Case Diagram

### Fig. 6.2: Use Case Diagram

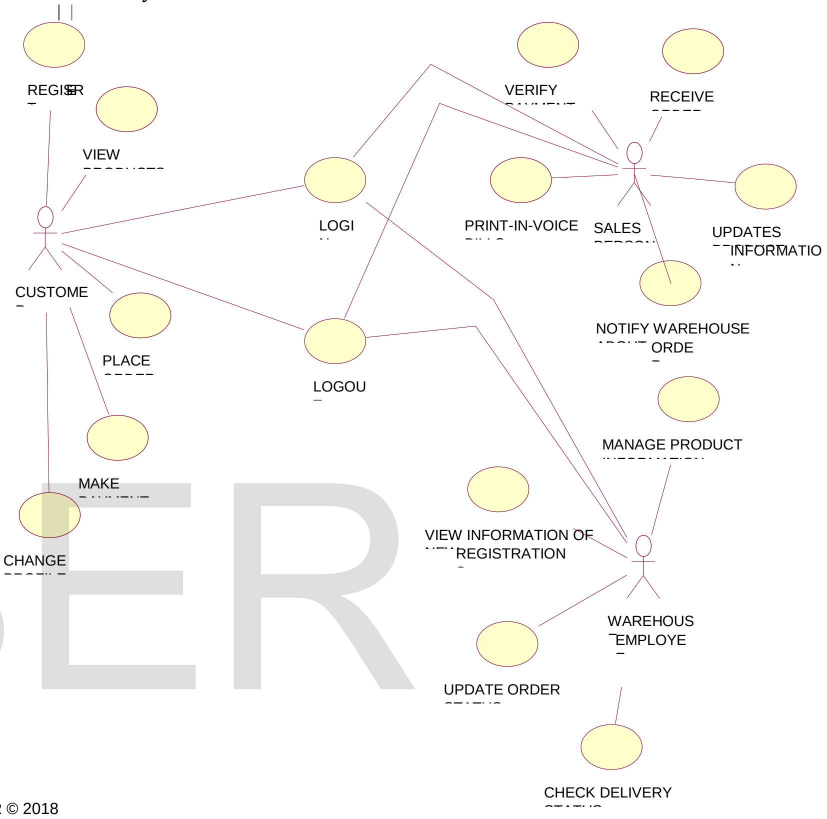
This use case diagram is a graphic depiction of the interactions among the users and the application of the system. The boundary defines the scope of the system.

### 6.3 System Requirements

#### 6.3.1 Hardware and Software

- **Hardware Requirement:** i3 Processor Based Computer
- 1 GB RAM
- 50 GB Hard Disk
- Monitor
- Internet Connection

- **Software Requirement**:
- Windows 7 or higher.
- WAMP Server Notepad++.
- My SQL 5.6.



### 6.4 Modules

This system comprises of 2 major modules and their submodules:

**1. Admin Login:** Admin need to login by providing the login credentials to access the below given admin modules.

o **Product Entry:-** Admin can enter details about new products details. o **View Order:-** Admin can view details about the order

placed by the user. o **View Users Details:-** Admin can view all the registered user's details.

**2. User Login/Registration:** User can register on the system and get his online account on site.

o **View Products:-** The products are arranged and can be viewed in categories.

o **Add to Cart:-** Users can add multiple products to cart.

o **Pay using Card:-** After total bill is calculated user can pay via credit card online.

o **View Order:-** User can view details about the order placed.

## 7 CONCLUSION

The project "Detecting Data Leaks via SQL Injection Prevention on an E-Commerce" is something like the original grocery shop shopping cart that is used by the customer in selecting certain products. Finally, after selection the customer confirms orders for all the purchasing items and submits his/her account details with tax information at the checkout counter.

This is sometimes a desirable feature in modern communication system architectures. AES encryption would allow the chaining together of different services without exposing the data to each of those services.

## REFERENCES

https://scholar.google.co.in/scholar?q=homomorphic+encryption&hl=en&as_sdt=0&as_vis=1&oi=scholart&sa=X&ved=0ahUKEwiBksqYo7zQAhUMRo8KHVOkBSIQgQMIFzAA

[1] Gregory T. Buehrer, Bruce W. Weide, and Paolo A. G. Sivilotti, in the research paper "Using Parse Tree Validation to Prevent SQL Injection Attacks" publishes there commendable work in 2005. The techniques for sql injection discovery seems impressive as this paper also covered the SQL parse tree validation very specifically which is mentionable.

[2] Zhendong Su and Gary Wassermann, in 2006 published their research in SQL injection entitled "The Essence of Command Injection Attacks in Web Applications" and covered the specific methods to check and sanitize input query using SQLCHECK, it use the augmented questions and SQLCHECK grammar to validate query.

[3] Stephen Thomas and Laurie Williams, worked so long in the field of SQL injection and in 2007, in their research paper, entitled "Automated Protection of PHP Applications against SQL-injection Attacks" they covered an authentic scheme to protect application automatically from SQL injection intrusion. This authentic approach combines static, dynamic analysis and intelligent code re-engineering to secure existing properties.

[4] Ke Wei, M. Muthuprasanna & Suraj Kothari in 2007, published their work entitled "Preventing SQL Injection Attacks in Stored Procedures" and through this they provides a novel approach to shield the stored procedures from attack and detect SQL injec-tion from websites. Their method includes runtime check with subsequent application code analysis to eliminate vulnerability to attack. The key method behind this vulnerability attack is that it modifies the composition of the original SQL statement and identifies the SQL injection attack.