Sri Lanka Institute of Information Technology

Bug Bounty Report - 03

Module: IE2062

Web Security

Year 2, Semester 2

Aazaf Ritha. J – IT23151710

B.Sc. (Hons) in Information Technology

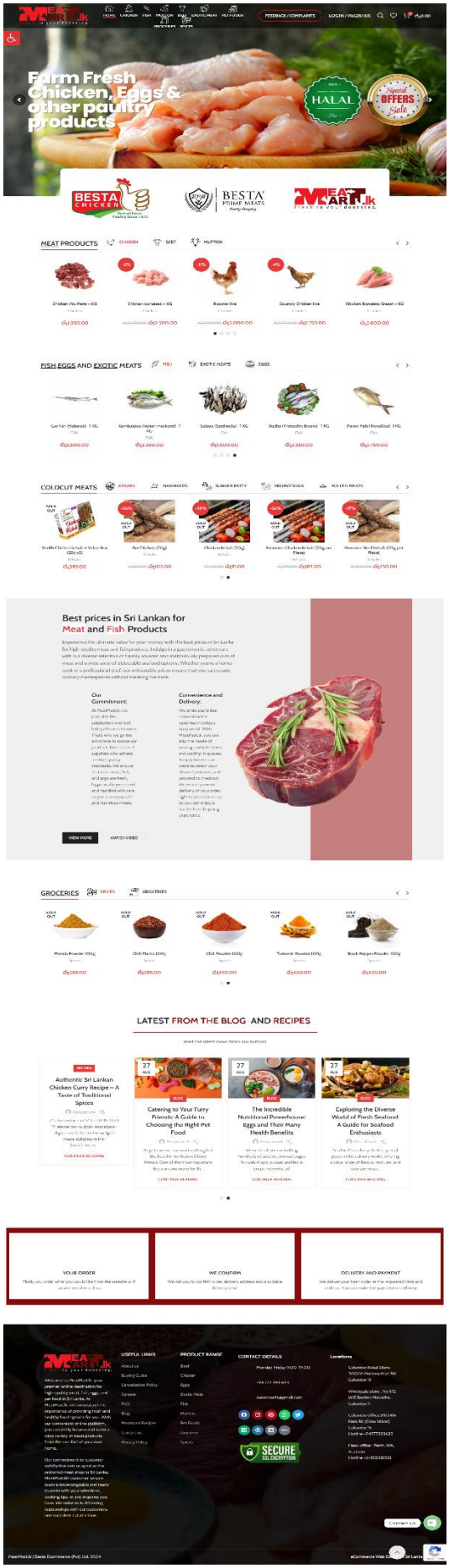
Specialized in Cyber Security

Table of Contents

Introduction 3
Vulnerability Title 5
Description 6
Affected Component 7
Impact Assessment 8
Steps to Reproduce 9
Proof of Concept (Screenshots) 10
Proposed Mitigation or Fix 13
Conclusion 14

Introduction

This report outlines a vulnerability discovered on a Sri Lankan e-commerce website, https://meatmart.lk. During security testing, a Clickjacking vulnerability was identified on multiple pages of the site. The issue was confirmed through both manual and automated methods, and this report provides technical details, reproduction steps, and mitigation recommendations.


Farm Fresh Chicken, Eggs & other paultry products
HALAL
Special OFFERS Sale
BESTA CHICKEN
BESTA FRESH MEATS
MEAT PRODUCTS
FISH EGGS AND EXOTIC MEATS
COLDCUT MEATS
Best prices in Sri Lankan for Meat and Fish Products
Our Commitment:
Convenience and Delivery:
VIEW MORE
WATCH VIDEO
GROCERIES
LATEST FROM THE BLOG AND RECIPES
Authentic Sri Lankan Chicken Curry Recipe – A Taste of Traditional Spices
Catering to Your Furry Friends: A Guide to Choosing the Right Pet Food
The Incredible Nutritional Powerhouse: Eggs and Their Many Health Benefits
Exploring the Diverse World of Fresh Seafood: A Guide for Seafood Enthusiasts
YOUR ORDER
WE CONFIRM
DELIVERY AND PAYMENT
USEFUL LINKS
PRODUCT RANGE
CONTACT DETAILS
Locations
SECURE SSL CERTIFICATE

Vulnerability Title

Title – Clickjacking Vulnerability

Risk Level- Medium

Domain - https://meatmart.lk/

Description

Clickjacking is a type of attack where a malicious website can trick a user into interacting with a page element (like a button or link) on a website that the user is unaware of. This can be done by embedding the legitimate website inside an iframe or another HTML element, which the attacker controls. The attacker may then "overlay" transparent or misleading elements over the embedded content, making it appear as if the user is clicking on a harmless interface, while in reality, they are triggering actions on the legitimate website. The vulnerability arises when a website allows itself to be embedded within an iframe without protections in place to prevent this behavior. In particular, Clickjacking attacks can exploit the absence of security headers like **X-Frame-Options** or **Content-Security-Policy (CSP)** that are designed to prevent websites from being embedded in iframes. The main risk of a clickjacking attack is that it can lead to unintended actions, such as unwitting users performing actions on behalf of the attacker (e.g., changing account settings, transferring money), or exposing sensitive data by tricking users into interacting with hidden elements. This vulnerability relates to **OWASP Top 10 A05:2021 – Security Misconfiguration**, as it stems from insufficient or missing security configurations that allow dangerous behaviors like iframe embedding, which should be explicitly disallowed.

Affected Component

Web Application: The vulnerability primarily affects web application or sites that allow their content to be embedded within an iframe without any restrictions.

/cart/

/blog/

/category/

/compare/

Browser Security Features: The vulnerability depends on whether the website implements the necessary security headers, such as **X-Frame-Options** or **Content-Security-Policy**.

Embedded Content (iframe): The use of **iframes** without proper security controls is the primary attack vector.

Impact Assessment

User Trust Erosion: If an attacker successfully carries out a clickjacking attack, it can undermine user trust in the affected website, as the user may unknowingly perform actions that harm their account or data.

Account Takeover: Clickjacking can lead to **unauthorized actions** like changing account settings, performing financial transactions, or enabling malicious configurations, which can lead to account takeovers or financial losses.

Loss of Sensitive Data: In some cases, the attacker might collect sensitive information if the victim interacts with the page unintentionally, leading to data breaches.

Reputation Damage: A website with clickjacking vulnerabilities can cause severe damage to its reputation and cause legal and compliance-related issues, especially if personal or financial data is involved.

Malicious Actions: Potential actions that can be performed unknowingly by the victim include:

- Clicking "Like" or "Share" on social media without realizing it.
- Unsubscribing from services.
- Modifying profile settings or making purchases.

Steps to Reproduce

Step 01: Identify the Target Website:

Identify the website that does not implement any clickjacking protection, particularly those that allow their pages to be embedded in an iframe.

- https://meatmart.lk/

Step 02: Create a Malicious Page:

Create an HTML page with an embedded iframe targeting the vulnerable site

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Clickjacking Attack</title>
</head>
<body>
<h2>Clickjacking Test</h2>
<button onclick="alert('You clicked the button!')">Click Me</button>
<iframe src="https://meatmart.lk/" width="800" height="400"></iframe>
</body>
</html>
```

Step 03: Trigger the Attack:

When a user visits the malicious page, they will see the "Click Me!" button. However, clicking the button will actually trigger a click event on the vulnerable website embedded in the iframe, such as making an unintended purchase or changing account settings.

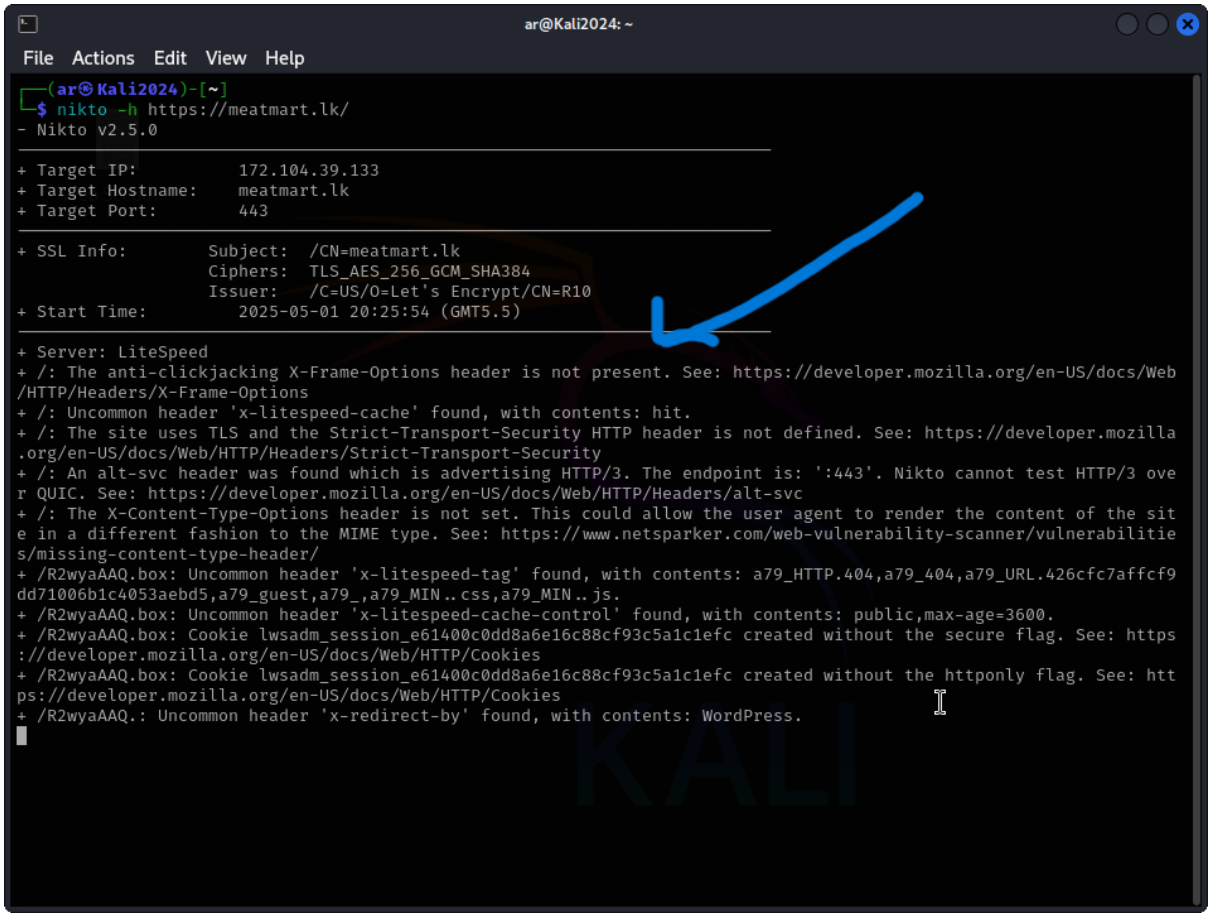
Step 04: Verify the Action:

Verify that the user performs the action on the underlying iframe (such as submitting a form or clicking a link) without realizing it.

Proof of Concept (Screenshots)

Nikto Scan

This screenshot shows the output of a Nikto scan against https://meatmart.lk. The scan reveals that the X-Frame-Options header is missing, confirming that the site does not implement basic protection against iframe-based attacks like Clickjacking.

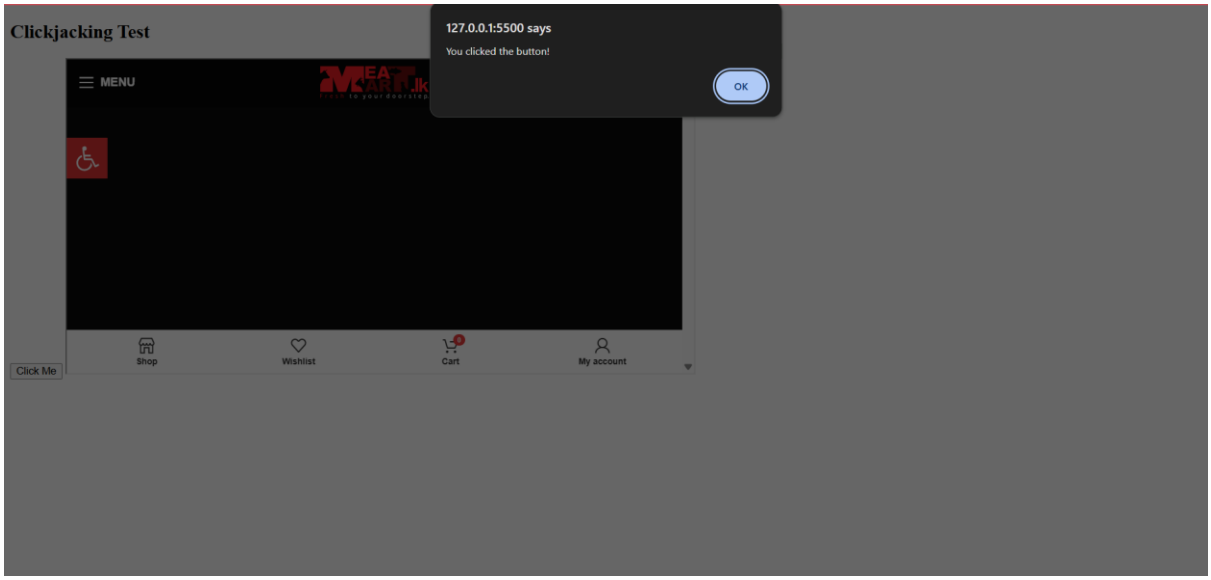


OWASP ZAP scan

The OWASP ZAP scan result highlights the absence of both X-Frame-Options and Content-Security-Policy headers. This indicates the website does not restrict where its content can be embedded, making it vulnerable to Clickjacking.

Manual Verification via iframe (Browser Screenshot)

This screenshot demonstrates a live test in which https://meatmart.lk is embedded inside an iframe on a separate domain. The site loads successfully, confirming that it can be iframed without restriction—validating the Clickjacking vulnerability.



Used Malicious code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Clickjacking Attack</title>
</head>
<body>
<h2>Clickjacking Test</h2>
<button onclick="alert('You clicked the button!')">Click Me</button>
<iframe src="https://meatmart.lk/" width="800" height="400"></iframe>
</body>
</html>
```

Proposed Mitigation or Fix

To mitigate the **Clickjacking** vulnerability, the following steps should be implemented:

Use X-Frame-Options Header:

This header can be used to prevent the website from being embedded in an iframe. Set the header to **DENY** or **SAMEORIGIN**:

- DENY: Prevents the page from being embedded in any iframe.
- SAMEORIGIN: Allows the page to be embedded only within iframes from the same origin.

Use Content-Security-Policy (CSP):

The **CSP** header provides an additional layer of protection. The frame-ancestors directive can be used to control where your content can be embedded.

JavaScript Clickjacking Protection:

Implement client-side defenses, such as using JavaScript to detect if the page is being loaded inside an iframe. If it is, you can force the top-level window to navigate to your site.

Test for Vulnerabilities:

Use security tools like **OWASP ZAP** or **Burp Suite** to test your website for potential clickjacking vulnerabilities. Additionally, running a **penetration test** periodically can help ensure that your site remains protected.

Conclusion

The Clickjacking vulnerability identified on https://meatmart.lk poses a moderate risk to users and the site's reputation. Without proper frame restrictions, attackers can exploit this flaw to trick users into performing unintended actions. This type of issue is associated with **"Security Misconfiguration,"** which is listed in the OWASP Top 10 most critical web application security risks. Immediate implementation of security headers like X-Frame-Options and Content-Security-Policy is recommended to mitigate this issue and improve the site's overall security posture.