Complete Web Application Testing using OWASP Vulnerabilities

24chakra.com

NAME: Nuthalapati Vipul

DATE: 03/11/2023

Email: vipulchowdary1@gmail.com

Table of Contents

Executive Summary

This project is a multifaceted exploration of cyber security, addressing five pivotal topics. It commences with an in-depth examination of SQL injection vulnerabilities and their variations, showcasing practical demonstrations on the 24Chakra web application. Classic, Error and Boolean based attacks are thoroughly investigated, underscoring the real-world consequences of these exploits.

The project then transitions to a holistic assessment of application security. It involves comprehensive testing with a specific focus on the OWASP 2021 Top 10 vulnerabilities. Detailed reports are generated, encompassing identified vulnerabilities, associated risks, and recommended mitigation strategies, thus contributing to strengthened application security practices.

Further, the project delves into the realm of mobile application security, culminating in a simulated phishing attack. This exercise underscores the tactics and vulnerabilities that malicious actors may leverage, highlighting the imperative for heightened mobile app security awareness.

Finally, the project embraces the bug bounty ecosystem. A security vulnerability is identified, responsibly disclosed, and submitted on the Bugcrowd platform, showcasing the symbiotic collaboration between security researchers and organizations to reinforce the security of digital assets.

In consolidating these five diverse facets of cyber security, this project serves as an educational and informative resource for a broad spectrum of stakeholders. Its insights contribute to a more resilient digital landscape and foster a culture of cyber security vigilance in the face of the ever-evolving cyber threat landscape.

Detailed Findings

HOST – 63.141.128.8

Name: Basic Pentesting
IP: 63.141.128.8

```
C:\Users\Hp>nslookup
Default Server:  reliance.reliance
Address:  2405:201:c026:388c::c0a8:1d01

> www.24chakra.com
Server:  reliance.reliance
Address:  2405:201:c026:388c::c0a8:1d01

Non-authoritative answer:
Name:    24chakra.com
Address:  63.141.128.8
Aliases:  www.24chakra.com
```

Created:	2018-11-13
Expires:	2024-11-13
Owner:	Redacted for Privacy (Privacy service provided by Withheld for Privacy ehf)
Hosting company:	Bigcommerce Inc.
Registrar:	NAMECHEAP, INC.
IPs:	63.141.128.8
DNS:	ns1.bigcommerce.com ns2.bigcommerce.com ns3.bigcommerce.com
Email:	See owner's emails

Last DNS records

	Record type	TTL	Value
	A	900	63.141.128.8
	CNAME	900	24chakra.com
+	MX	3600	ALT4.ASPMX.L.GOOGLE.com
+	MX	3600	ALT3.ASPMX.L.GOOGLE.com
+	MX	3600	ASPMX.L.GOOGLE.com
+	MX	3600	ALT2.ASPMX.L.GOOGLE.com
+	MX	3600	ALT1.ASPMX.L.GOOGLE.com
	NS	900	ns3.bigcommerce.com
	NS	900	ns2.bigcommerce.com
	NS	900	ns1.bigcommerce.com

Last HTTPS Certificate

JARM Fingerprint

29d3dd00029d29d00042d43d00041d5de67cc9954cc85372523050f20b5007

Last HTTPS Certificate

```
Data:
   Version: V3
   Serial Number: f5a87f09065a9cf9854a1d3e2a3e99f
   Thumbprint: cb396ad6eff2a8baddf56d4d449796f9333782b1
```

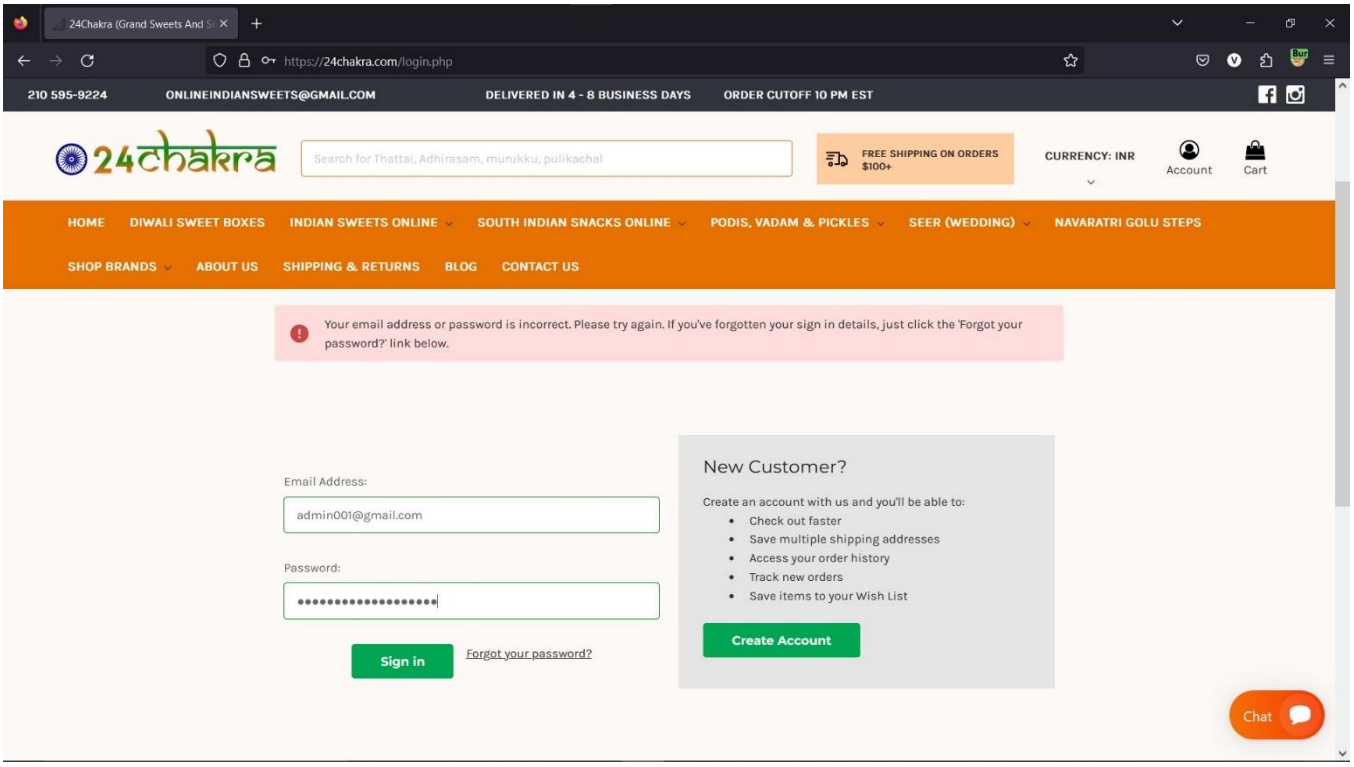
❖ SQL Injection (3 types) - LOW

Description

SQL injection is a type of security vulnerability and attack that targets databases used in web applications. It occurs when an attacker inserts or manipulates malicious SQL payload into input fields that are later executed by the application's database.

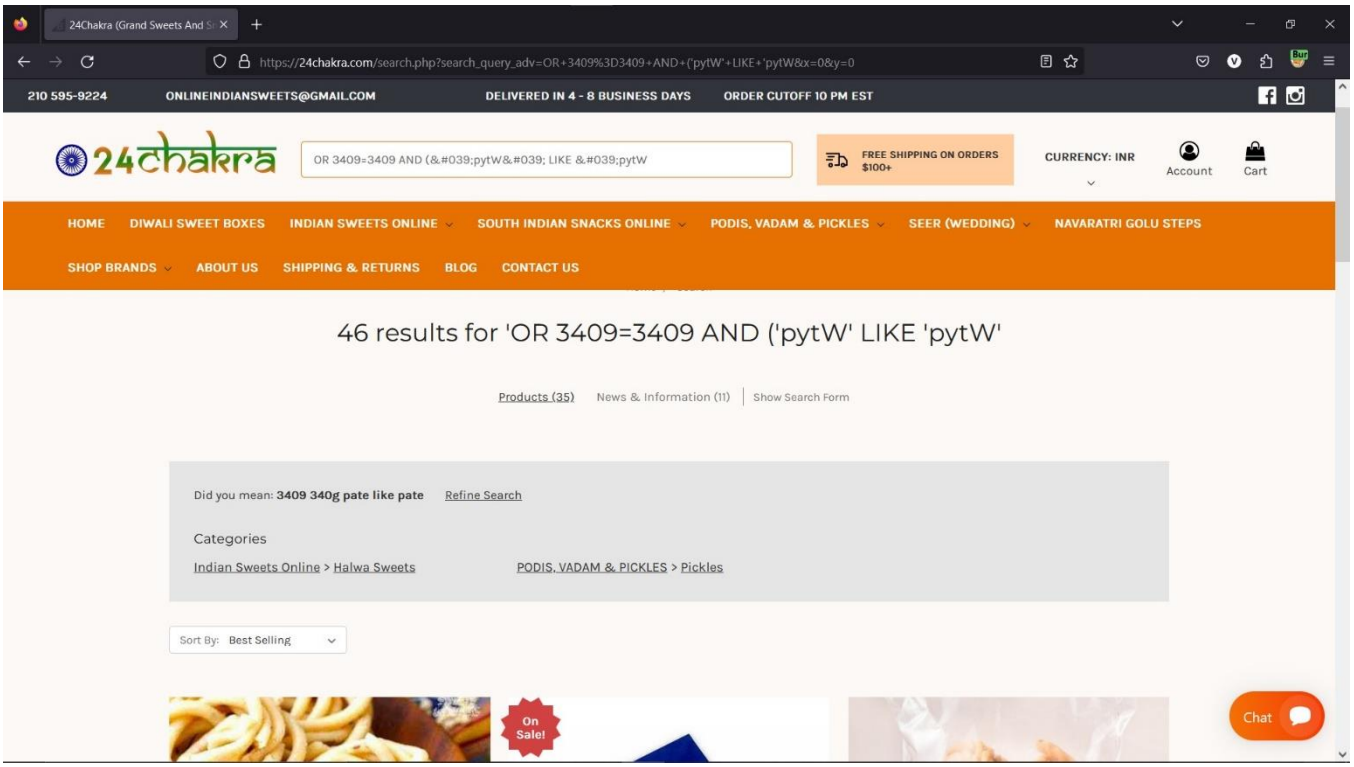
Analysis

I. **Classic SQL Injection:** This involves inserting malicious SQL code directly into input fields to manipulate database queries.



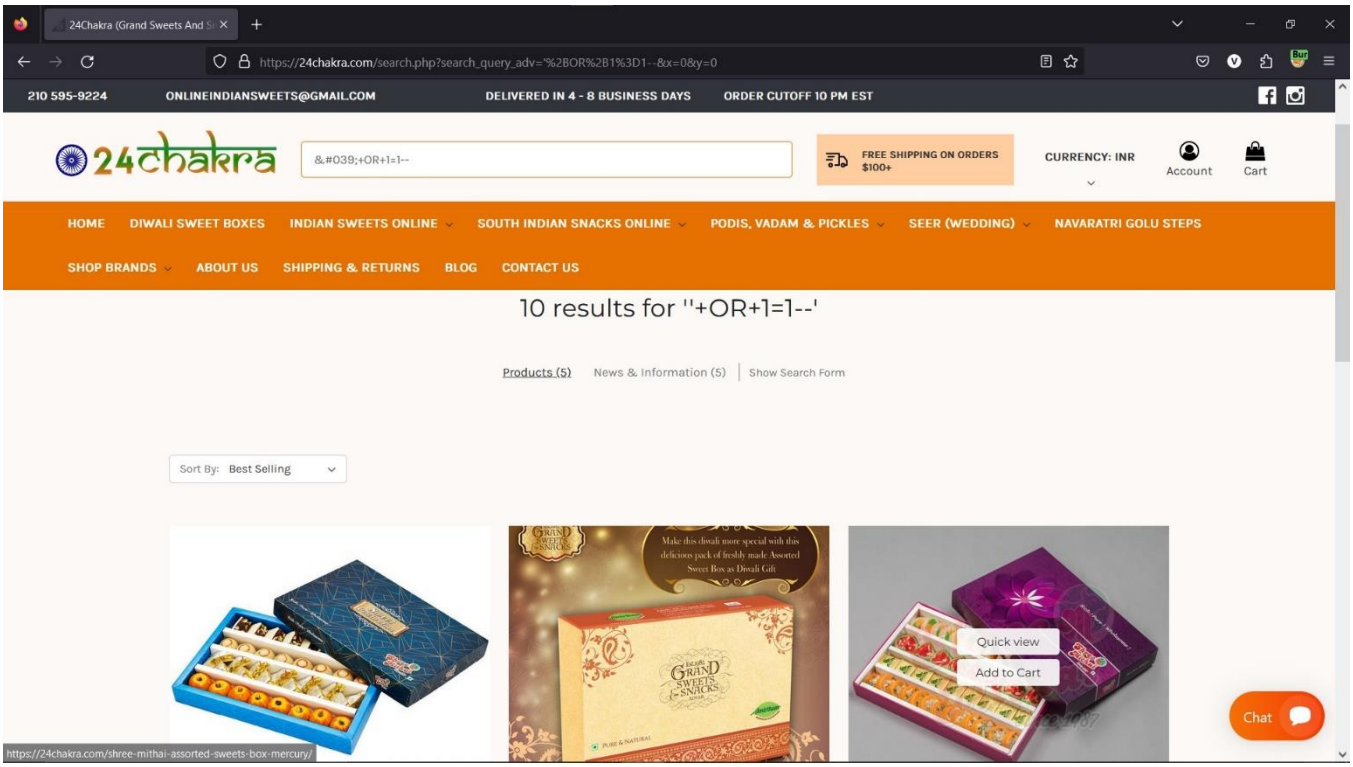
Classic SQL injection is not possible on this website to bypass the authentication.

II. **Error-Based SQL Injection:** It is a type of security vulnerability and attack that occurs in web applications and database-driven systems. This form of SQL injection is characterized by the attacker intentionally triggering SQL errors to gain information about the structure and content of a database.



Error-Based SQL payload is inserted in search bar and forwarded the request, but this type of SQL injection is also not possible on this website.

III. **Boolean-Based SQL Injection:** This is another type of injection attack that relies on the use of Boolean logic to infer information about a database. Instead of extracting data directly, the attacker manipulates the application to make true or false statements.



Boolean based payloads also not possible in this website.

This website is secured against SQL injection that has implemented robust input validation and parameterization techniques, significantly reducing the risk of malicious database manipulation through user input.

Remediation

Option 1: Use Parameterized Statements or Prepared Statements- Implement parameterized queries or prepared statements to ensure that user input is treated as data, not as executable SQL code.

Option 2: Input Validation- Validate and sanitize user input to block potentially malicious input.

Option 3: Principle of Least Privilege- Ensure that the database user account used by the application has the minimum necessary privileges.

❖ Broken Authentication – LOW

Description

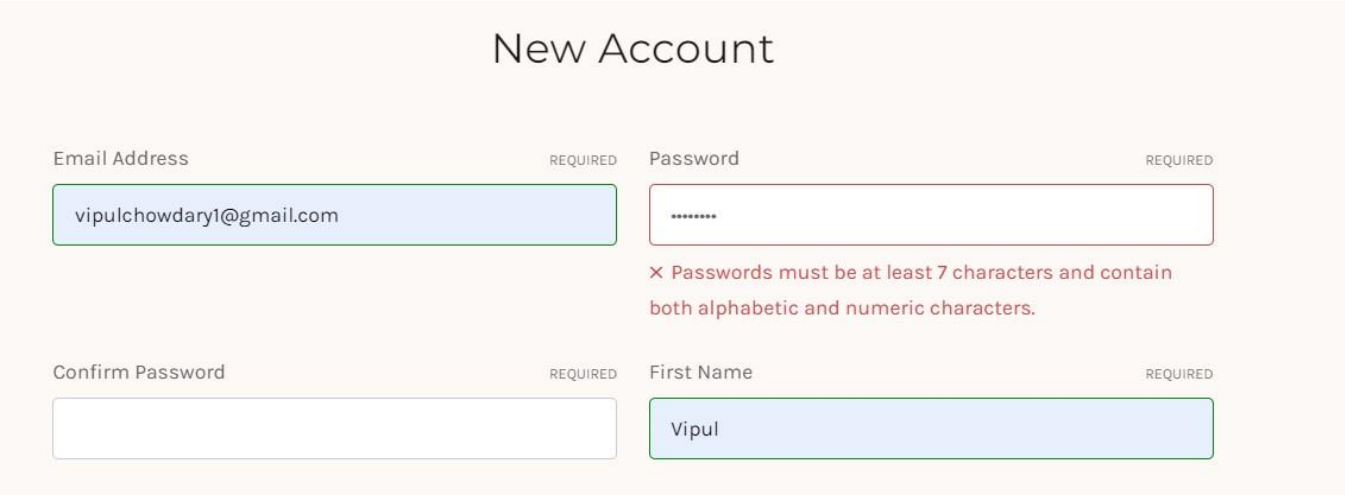
Broken Authentication is a security vulnerability that occurs when an attacker exploits weaknesses in an application's user authentication mechanisms. It can lead to unauthorized access to user accounts, data or system functionality due to inadequate or flawed authentication processes.

Analysis

Weak password policies: Insufficient password complexity requirements make it easier for attackers to guess or crack password.

Flawed Authentication Procedures: Insecure login mechanisms or insufficient multi-factor authentication (MFA) can lead to vulnerabilities.

Credentials Storage: Storing credentials in an insecure manner, such as plain text or weakly hashed passwords, is a common cause.



This website contains better password policy, password must be in alphanumeric which gives better protection.

Remediation

Option 1: Strong Password Policies- Enforce strong password policies, including complexity requirements and periodic password changes.

Option 2: Session Management- Implement secure session management practices, including session timeout and secure token handling.

Option 3: Multi-Factor Authentication (MFA) - Encourage or require the use of MFA to enhance authentication security

❖ Sensitive Data Exposure – HIGH

Description

Sensitive data exposure is a security risk where confidential or private information is inadvertently disclosed or accessed by unauthorized parties. It occurs when systems or applications fail to adequately protect sensitive data, as a result, it can be exposed to hackers or other malicious actors. This vulnerability can lead to data breaches, identity theft, underscoring the importance of robust encryption, access controls and data security measures to safeguard sensitive information.

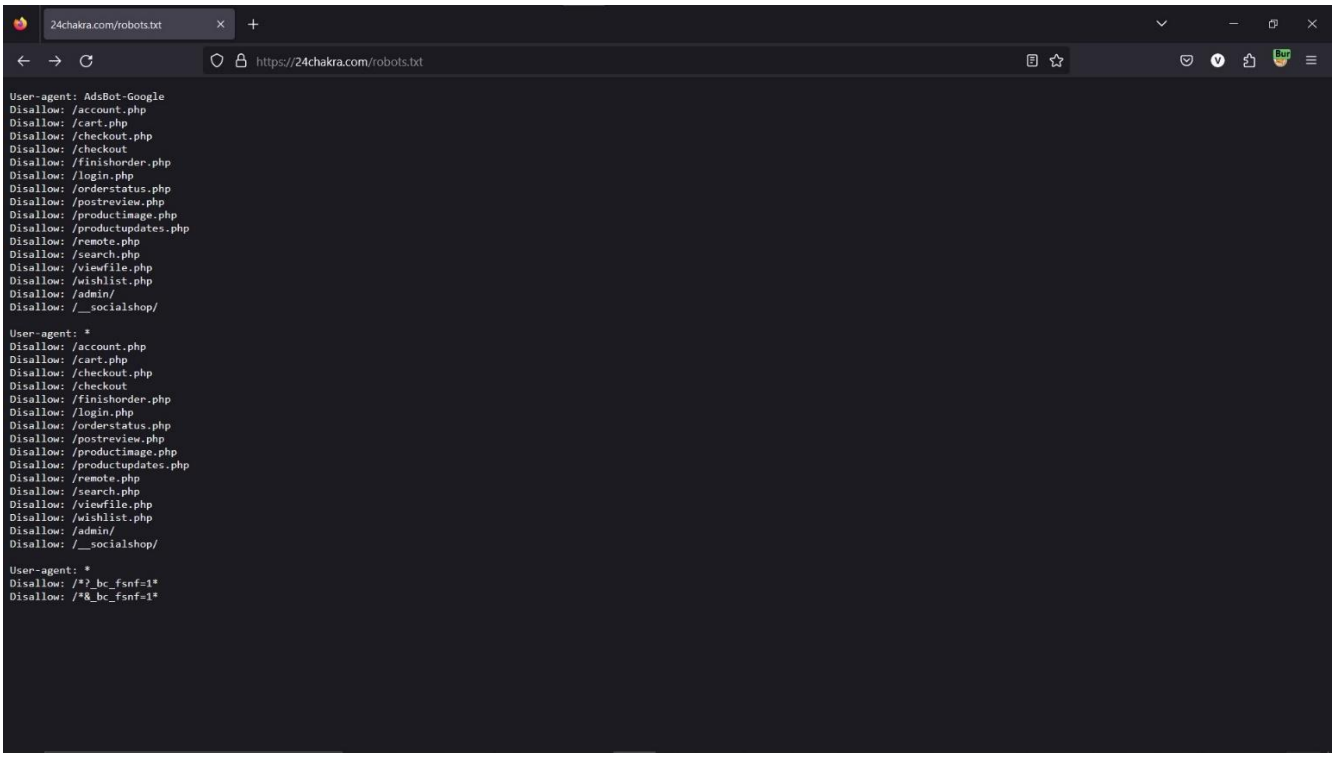
Analysis

The web server contains a robots.txt file.

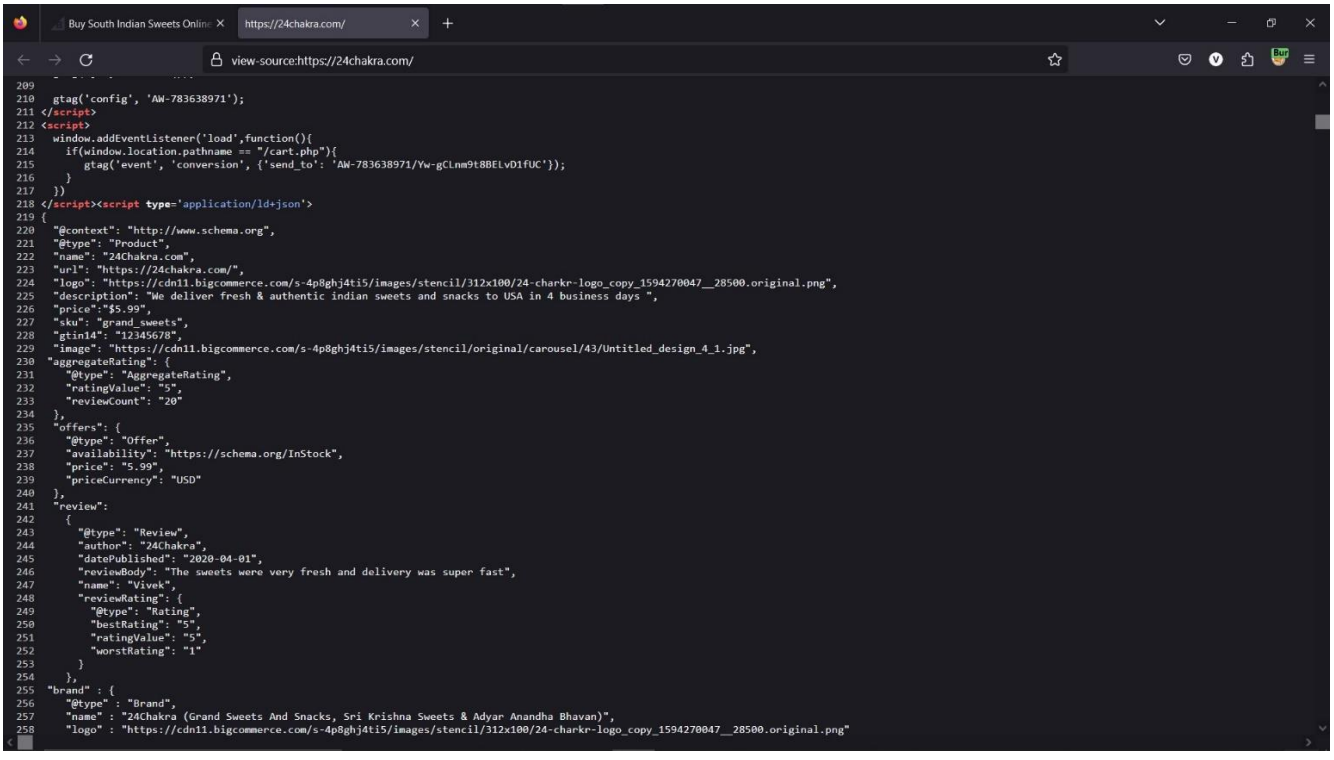
The file robots.txt is used to give instructions to web robots, such as search engine crawlers, about locations within the web site that robots are allowed, or not allowed, to crawl and index.

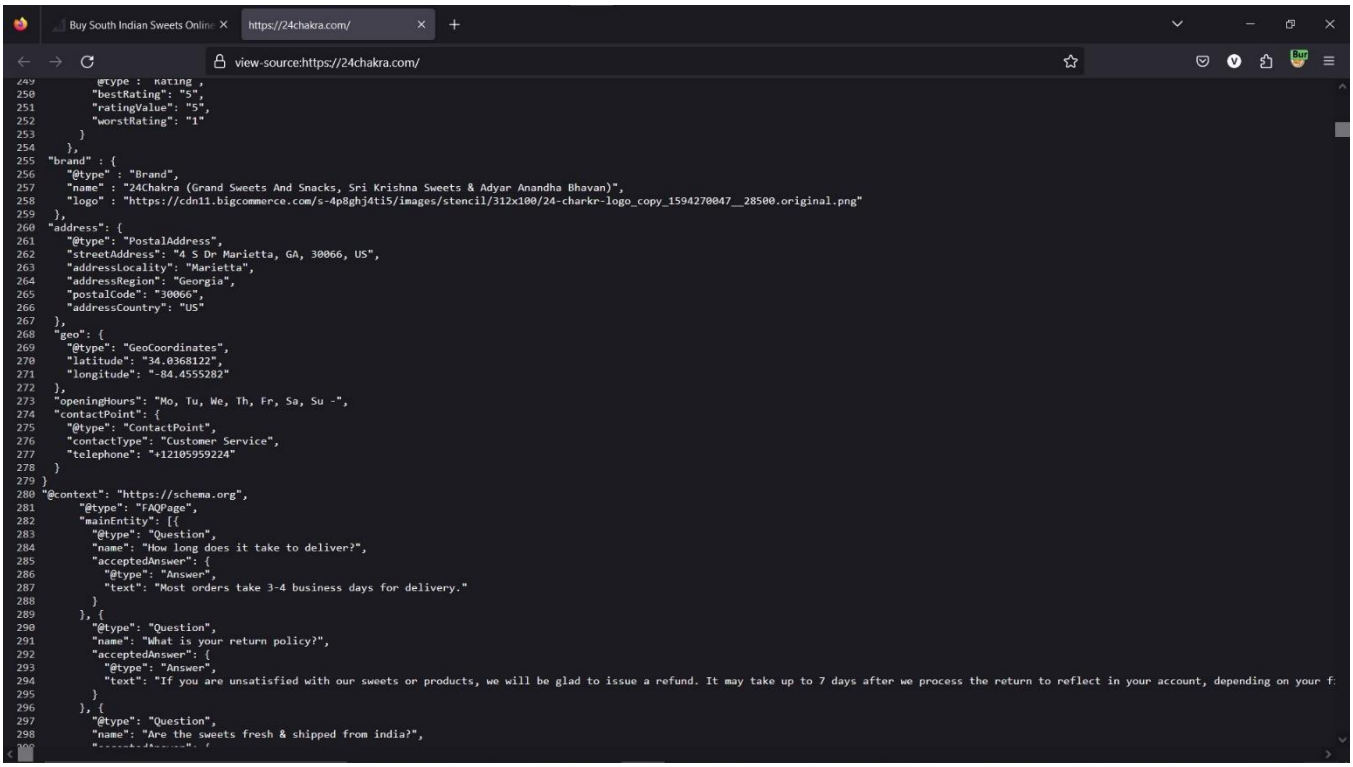
The presence of the robots.txt does not in itself present any kind of security vulnerability. However, it is often used to identify restricted or private areas of a site's contents. The information in the file may therefore help an attacker to map out the site's contents, especially if some of the locations identified are not linked from elsewhere in the site.

If the application relies on robots.txt to protect access to these areas, and does not enforce proper access control over them, then this presents a serious vulnerability



The source code is not encrypted, which exposes detailed information about the website's inner workings and customers details and their address.





Remediation

Option 1: Encryption- Encrypt sensitive data at rest and in transit using strong encryption algorithms.

Option 2: Access Controls- Implement strict access controls and authentication mechanisms to restrict access to sensitive data.

Option 3: Data Classification- Clearly identify and classify sensitive data and apply security controls accordingly.

❖ Broken Access Control – LOW

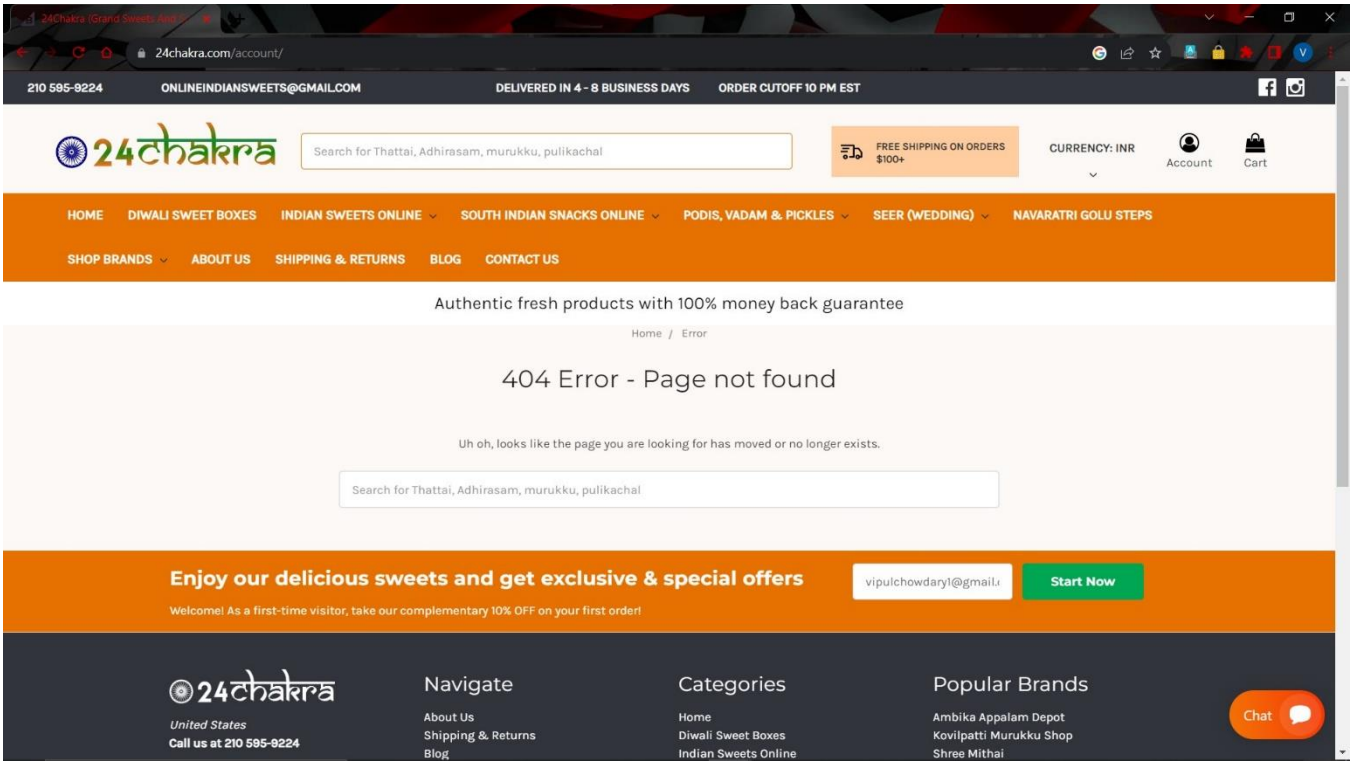
Description

Broken Access Control Vulnerability arise when an application fails to enforce proper access controls on resources. This can lead to unauthorized users gaining access to sensitive functionality or data. It's essential to implement granular access control policies and thoroughly test them to prevent these vulnerabilities.

Analysis

Data Exposure: Broken Access control can result in unauthorized users viewing sensitive information, potentially leading to data breaches.

Unauthorized Actions: Attackers may be able to perform actions they shouldn't, such as changing user roles, altering data or deleting records.



This is the web page when we tried to gain access by modifying URL, it is displaying 404 error.

The user session is encrypted and unable to gain access easily.

This website is secured according to broken access control.

Remediation

Option 1: Implement Proper Access Controls- Enforce access controls and authorization checks at the application level.

Option 2: Role-Based Access Control (RBAC) - Use RBAC to assign and manage user roles and permissions.

Option 3: Regular Testing- Conduct regular security testing to identify and rectify access control issues.

- **Insecure Design –** LOW

Description

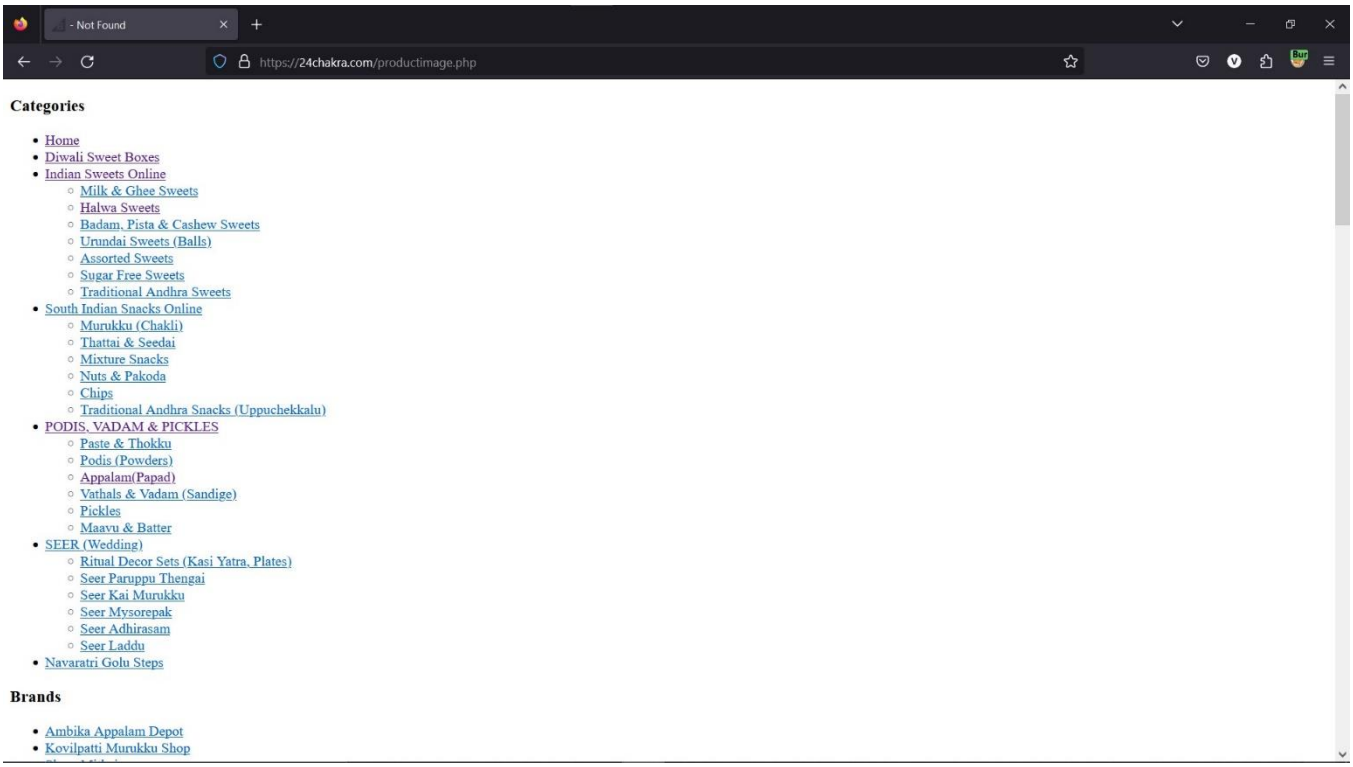
Insecure design is a new category in the OWASP Top 10 in 2021. It is a broad category related to critical design and architectural flaws in web applications that hackers can exploit.

Insecure designs can't be fixed by a perfect implementation. They require security controls to mitigate the threats.

Analysis

If an application trusts an HTTP request header like X-Forwarded-For to accurately specify the remote IP address of the connecting client, then malicious clients can spoof their IP address. This behavior does not necessarily constitute a security vulnerability, however some applications use client IP addresses to enforce access controls and rate limits. For example, an application might expose administrative functionality only to clients connecting from the local IP address of the server, or allow a certain number of failed login attempts from each unique IP address. Consider reviewing relevant functionality to determine whether this might be the case.

The application may be vulnerable to DOM-based client-side JSON injection. Data is read from document.cookie and passed to JSON.parse.



This website design needs to be fixed in this area to get better security controls.

Remediation

Option 1: Fix in design phase

Option 2: Threat modeling

Option 3: Implement security controls

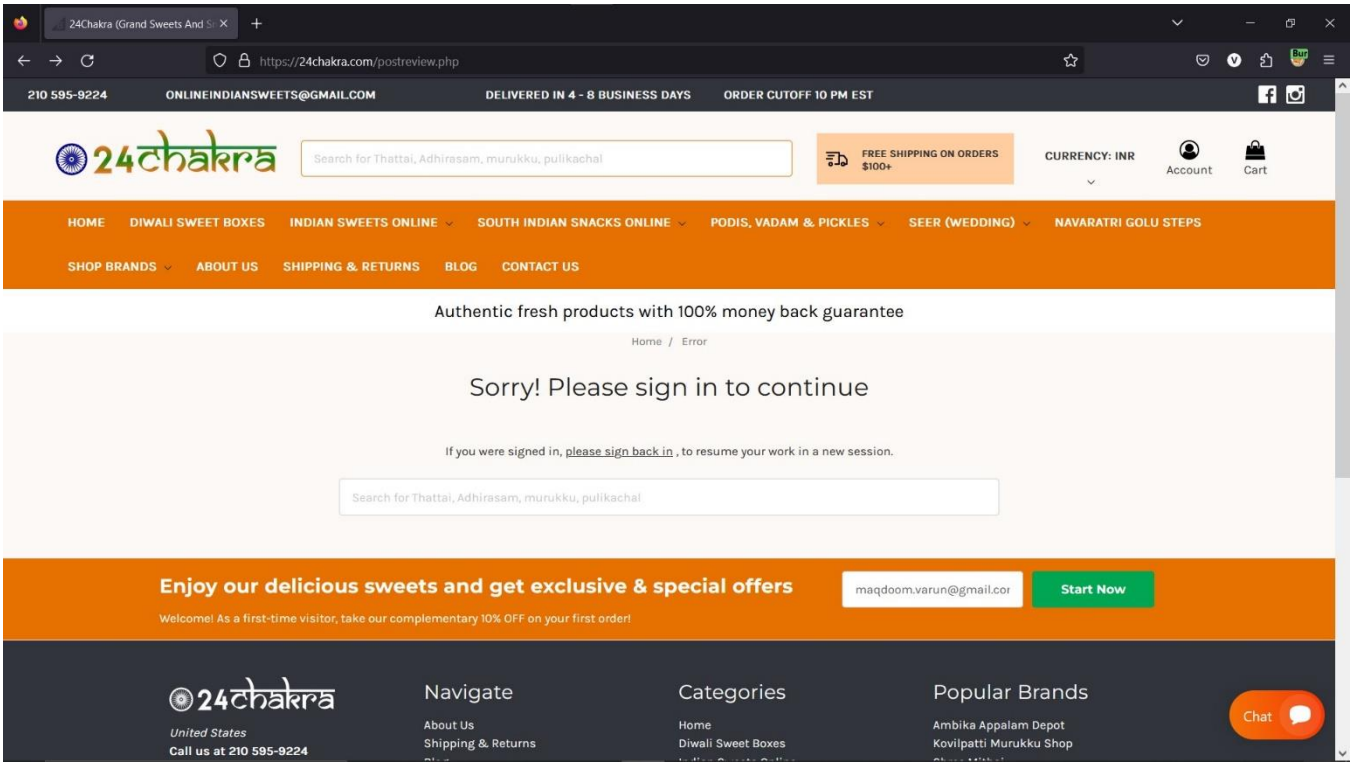
❖ Security Logging and Monitoring Failures – MEDIUM

Description

Security Logging and Monitoring Failures occur when an organization's systems, networks or applications lack the capability to log relevant security events, or when the monitoring of these logs is inadequate. This includes failure in capturing, storing, analyzing and responding to security related data.

Analysis

Inadequate Logging: Insufficient or incomplete logging of security-relevant events, leaving gaps in the audit trail.
Ignoring Critical Events: Failure to prioritize and log critical security events, leading to the potential oversight of malicious activities.
Poor Log Management: Lack of effective log management practices, including retention, backup, and secure storage.
Limited Analysis: Insufficient monitoring and analysis tools to identify and respond to suspicious or malicious activities.



This website is asking to sign in whenever we tried to open post review. There is Logging issue in this area.

Remediation

Option 1: Comprehensive Logging- Ensure that all relevant security events are logged, including authentication attempts, access control changes, and system activities.
Option 2: Effective Monitoring- Employ security information and event management (SIEM) systems and intrusion detection systems (IDS) to monitor and analyze logs in real-time.
Option 3: Alerting and Incident Response- Establish well-defined incident response procedures, including immediate alerts for suspicious activities.
Option 4: Log Retention and Backup- Implement secure log storage, backup, and retention policies to comply with legal requirements and forensic analysis.

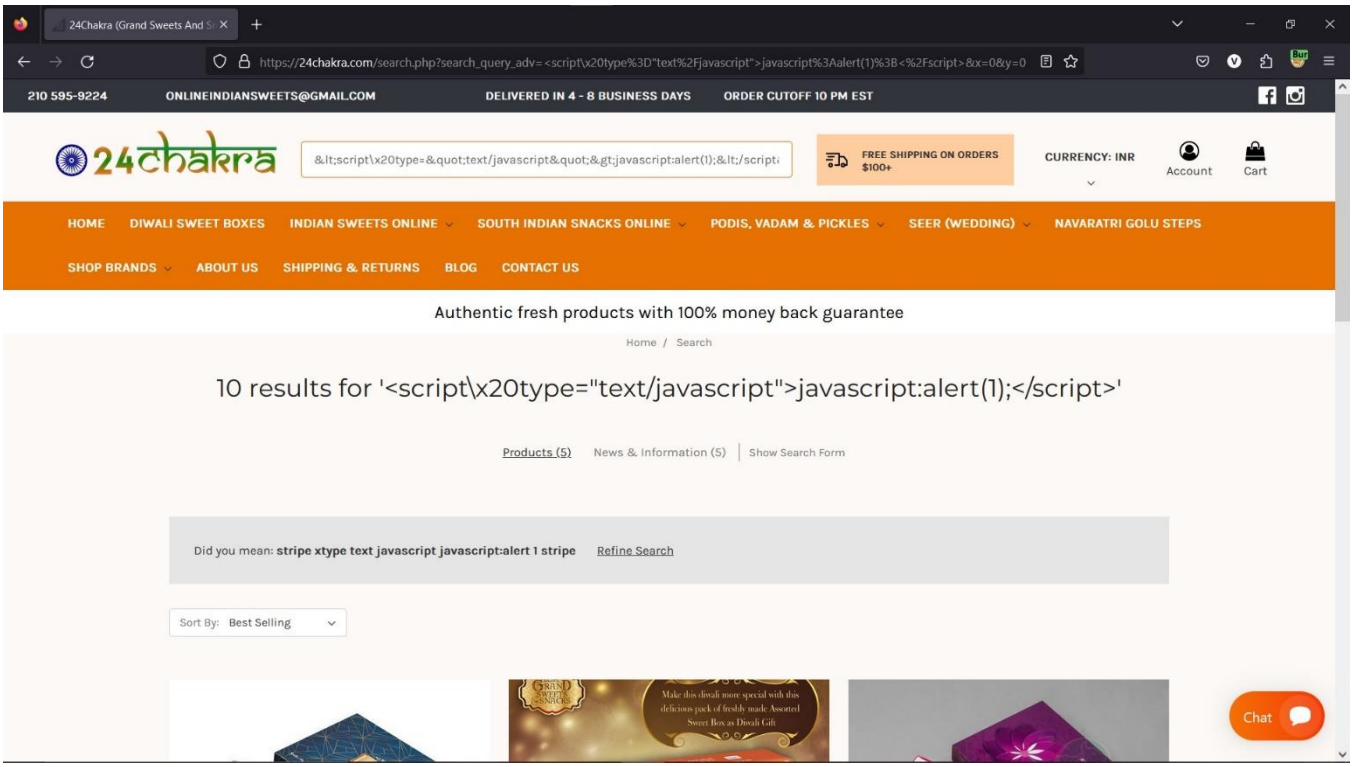
❖ Cross-Site Scripting (XSS) – LOW

Description

XSS vulnerabilities occur when an attacker injects malicious scripts into web pages viewed by other users. These scripts can execute in the context of the user's browser, potentially stealing sensitive information, hijacking sessions, or defacing websites. Web developers need to validate and sanitize user input and use output encoding to mitigate XSS vulnerabilities.

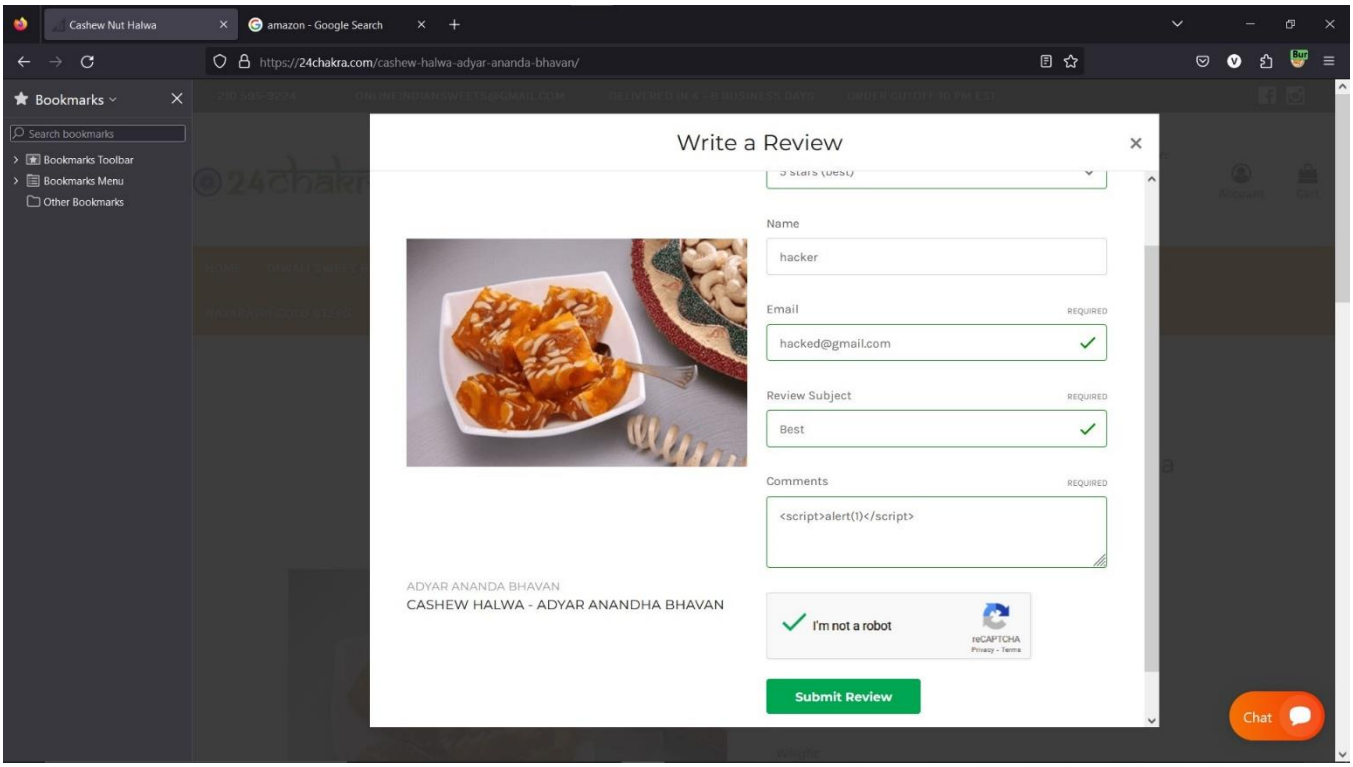
Analysis

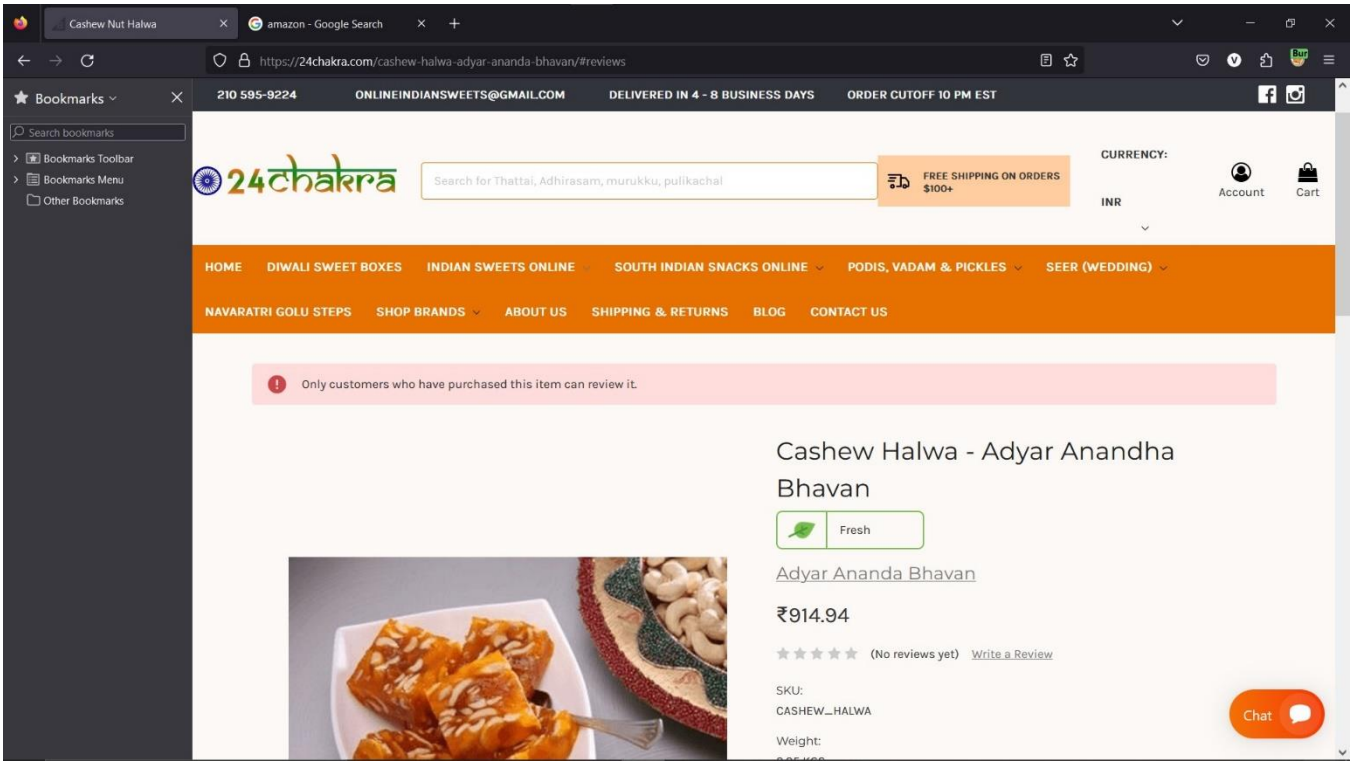
I. **Reflected Cross-Site Scripting (XSS):** Reflected Cross-Site Scripting (XSS) is a type of web application security vulnerability where an attacker injects malicious scripts into a web application, and these scripts are then immediately reflected and executed in the response to a user's request. This type of XSS attack is often used to target a specific user, as the injected script is included in the response generated for that user and executed in their browser.



Reflected XSS is not possible in this website.

II. **Stored Cross-Site Scripting (XSS):** Stored Cross-Site Scripting (XSS) is another type of web application security vulnerability where an attacker injects malicious scripts into a web application, and these scripts are stored on the server. When other users access the affected web page, the stored script is served from the server, leading to an XSS attack.


Cashew Nut Halwa
amazon - Google Search
https://24chakra.com/cashew-halwa-adyar-ananda-bhavan/
Bookmarks
Search bookmarks
Bookmarks Toolbar
Bookmarks Menu
Other Bookmarks
Write a Review
Name
hacker
Email
REQUIRED
hacked@gmail.com
Review Subject
REQUIRED
Best
Comments
REQUIRED
<script>alert(1)</script>
ADYAR ANANDA BHAVAN
CASHEW HALWA - ADYAR ANANDHA BHAVAN
I'm not a robot
reCAPTCHA
Submit Review
Chat


Cashew Nut Halwa
amazon - Google Search
https://24chakra.com/cashew-halwa-adyar-ananda-bhavan/#reviews
Bookmarks
Search bookmarks
Bookmarks Toolbar
Bookmarks Menu
Other Bookmarks
210 595-9224
ONLINEINDIANSWEETS@GMAIL.COM
DELIVERED IN 4 - 8 BUSINESS DAYS
ORDER CUTOFF 10 PM EST
24chakra
Search for Thattai, Adhirasam, murukku, pulikachal
FREE SHIPPING ON ORDERS $100+
CURRENCY:
INR
Account
Cart
HOME
DIWALI SWEET BOXES
INDIAN SWEETS ONLINE
SOUTH INDIAN SNACKS ONLINE
PODIS, VADAM & PICKLES
SEER (WEDDING)
NAVARATRI GOLU STEPS
SHOP BRANDS
ABOUT US
SHIPPING & RETURNS
BLOG
CONTACT US
Only customers who have purchased this item can review it.
Cashew Halwa - Adyar Anandha Bhavan
Fresh
Adyar Ananda Bhavan
₹914.94
(No reviews yet)
Write a Review
SKU:
CASHEW_HALWA
Weight:
Chat

This website is not allowing users to comment or review unless user purchases the particular product. This website is validating input and verifying purchases.

Stored XSS is not possible for this website.

Remediation

Option 1: Input Validation- Validate and sanitize user input, including output encoding to prevent XSS attacks.

Option 2: Content Security Policy (CSP) - Implement a CSP to restrict the sources of content that a web page can load.

Option 3: Output Encoding- Encode user-generated content before rendering it in the browser.

❖ Identification and Authentication Failure –HIGH

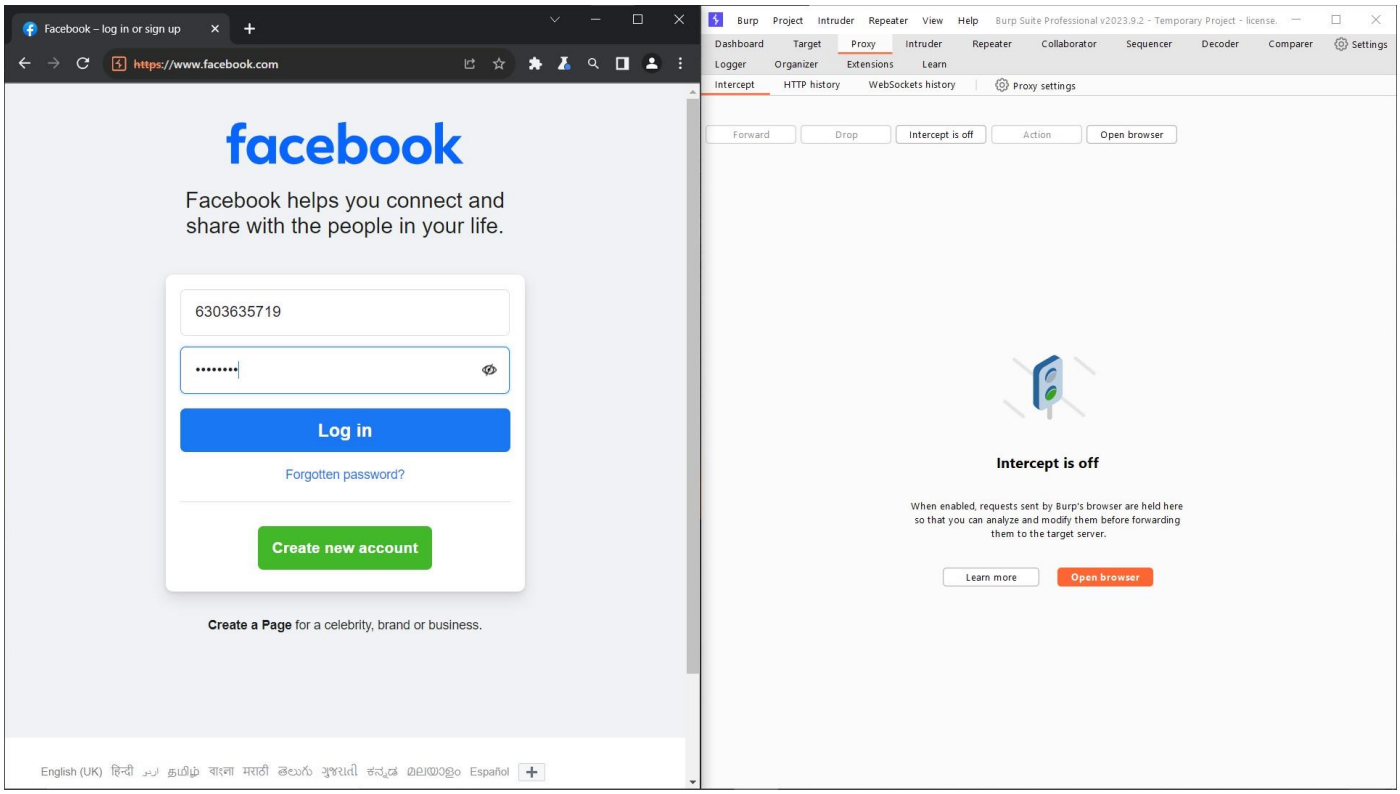
Description

Identification and Authentication Failure is a critical concern in cyber security. It refers to the breakdown in verifying the identities of users or systems trying to access sensitive data or resources. Such failures can lead to unauthorized access, data breaches, and security threats. In this discussion, we'll delve into the challenges, causes, and solutions related to identification and authentication failure to help you better protect your digital assets and maintain the integrity of your systems.

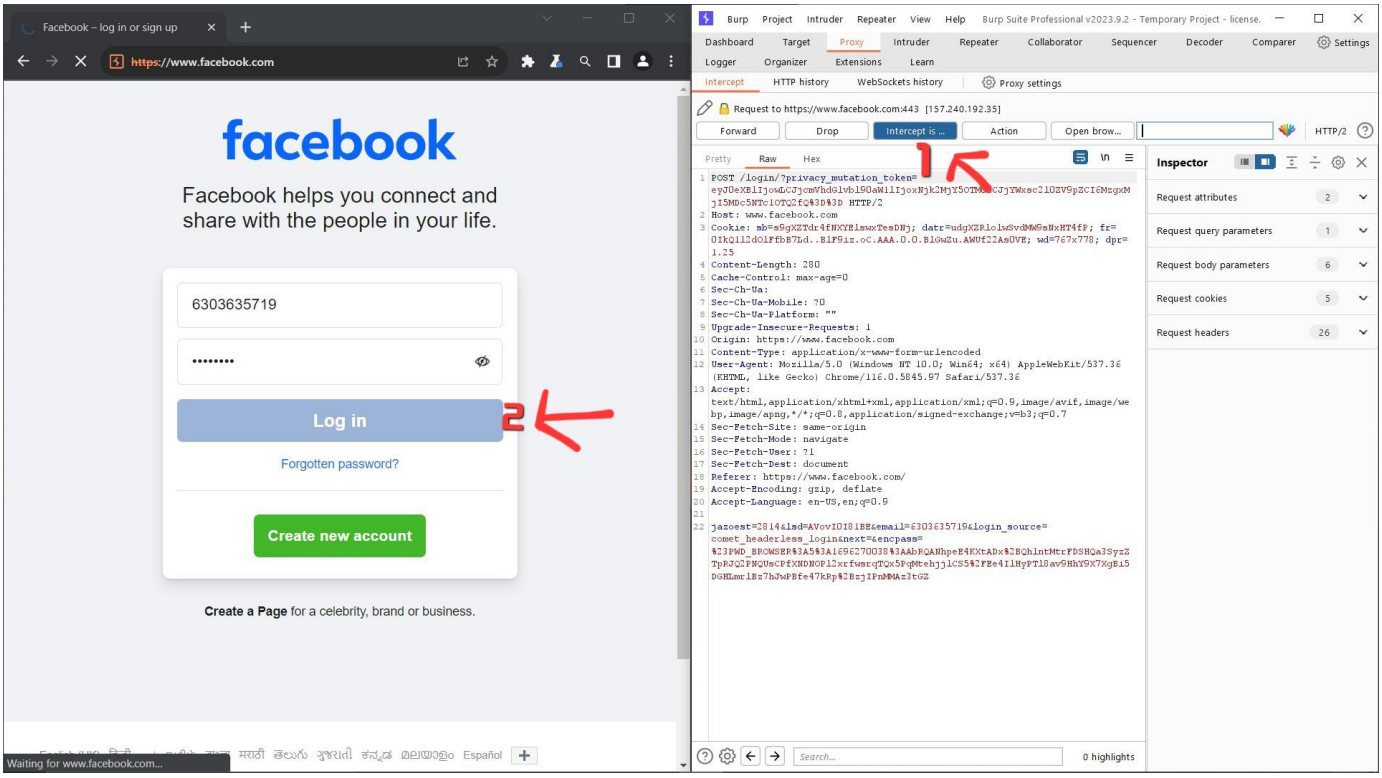
Analysis

Password cracking by brute force attack:

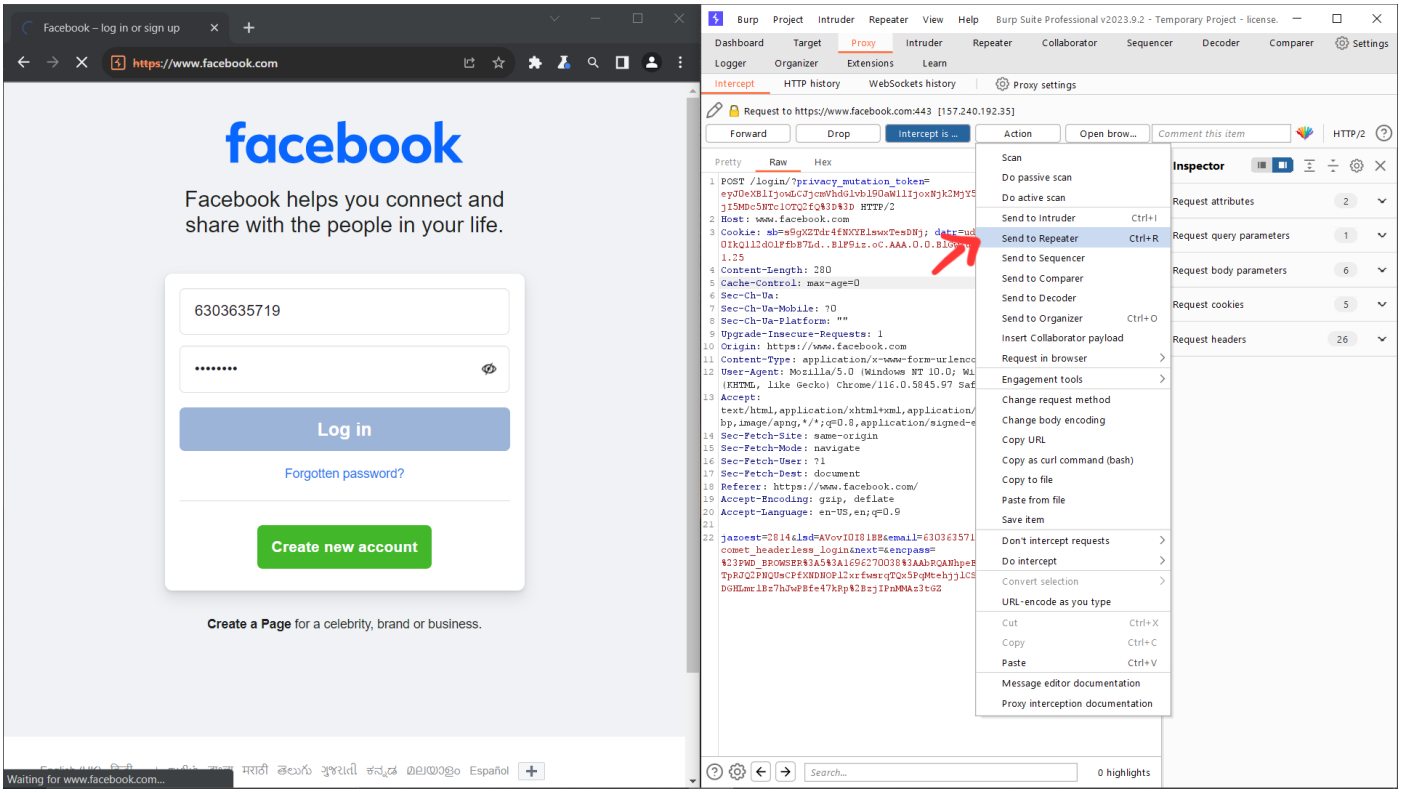
- Start Burp suite and Browser side by side, open website login page (example Facebook).



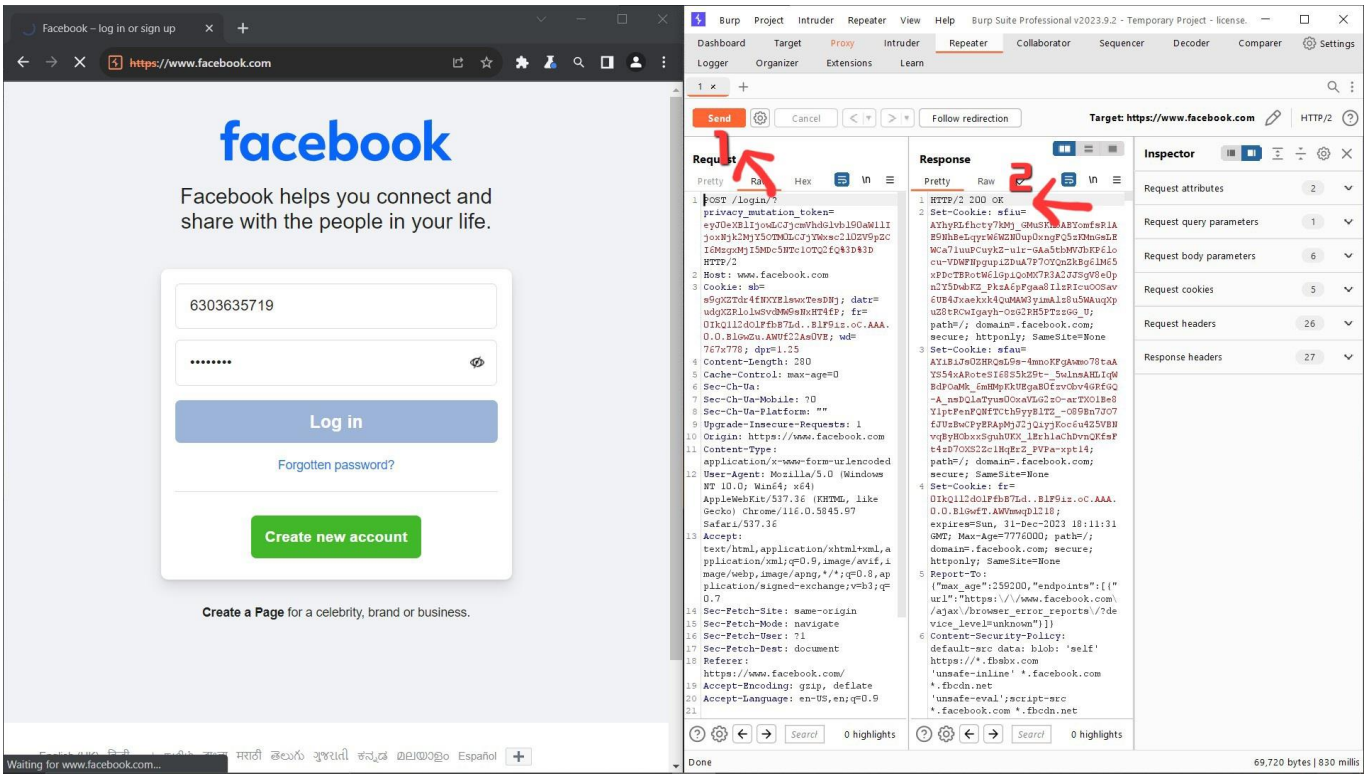
- Enter username of target and enter random password.



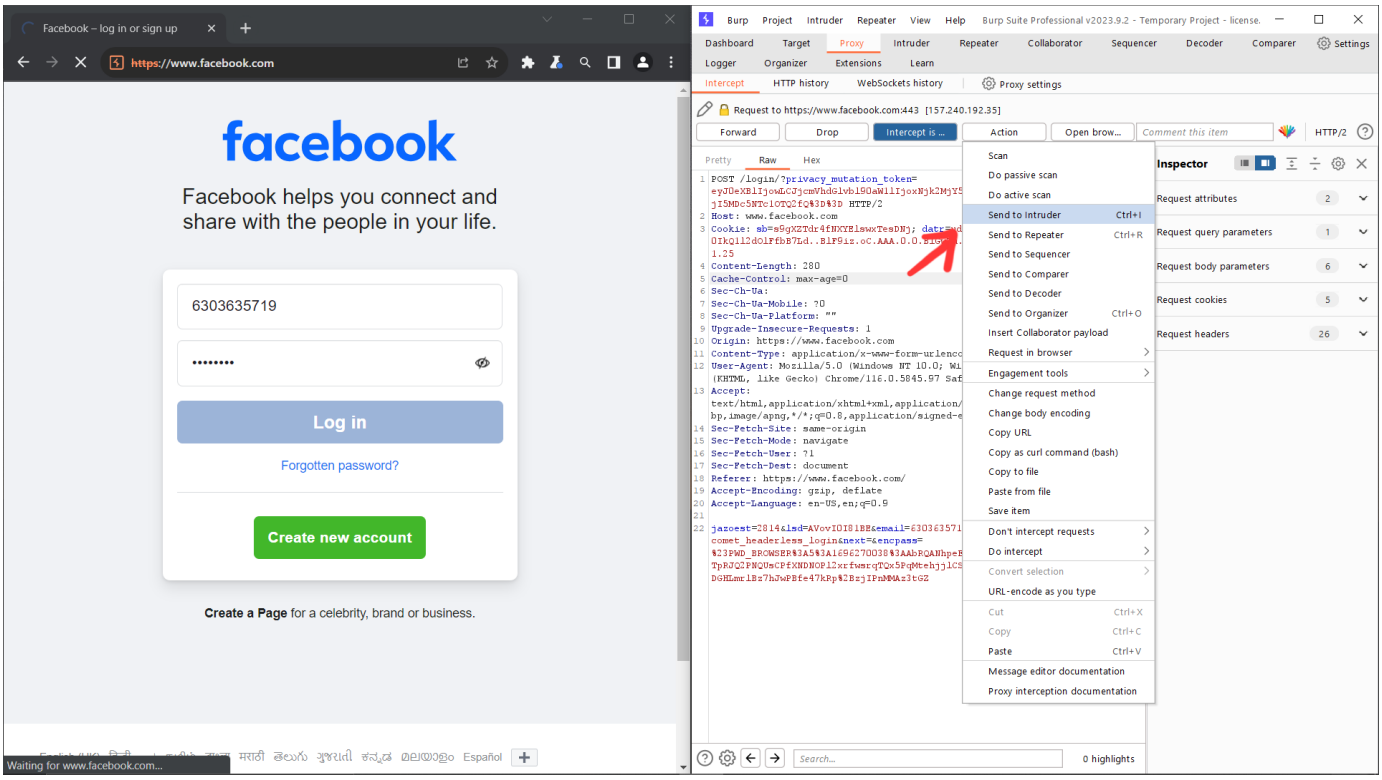
- Before clicking "Log in", turn interception ON in burp suite (like 1 in above pic) and then click login (2).



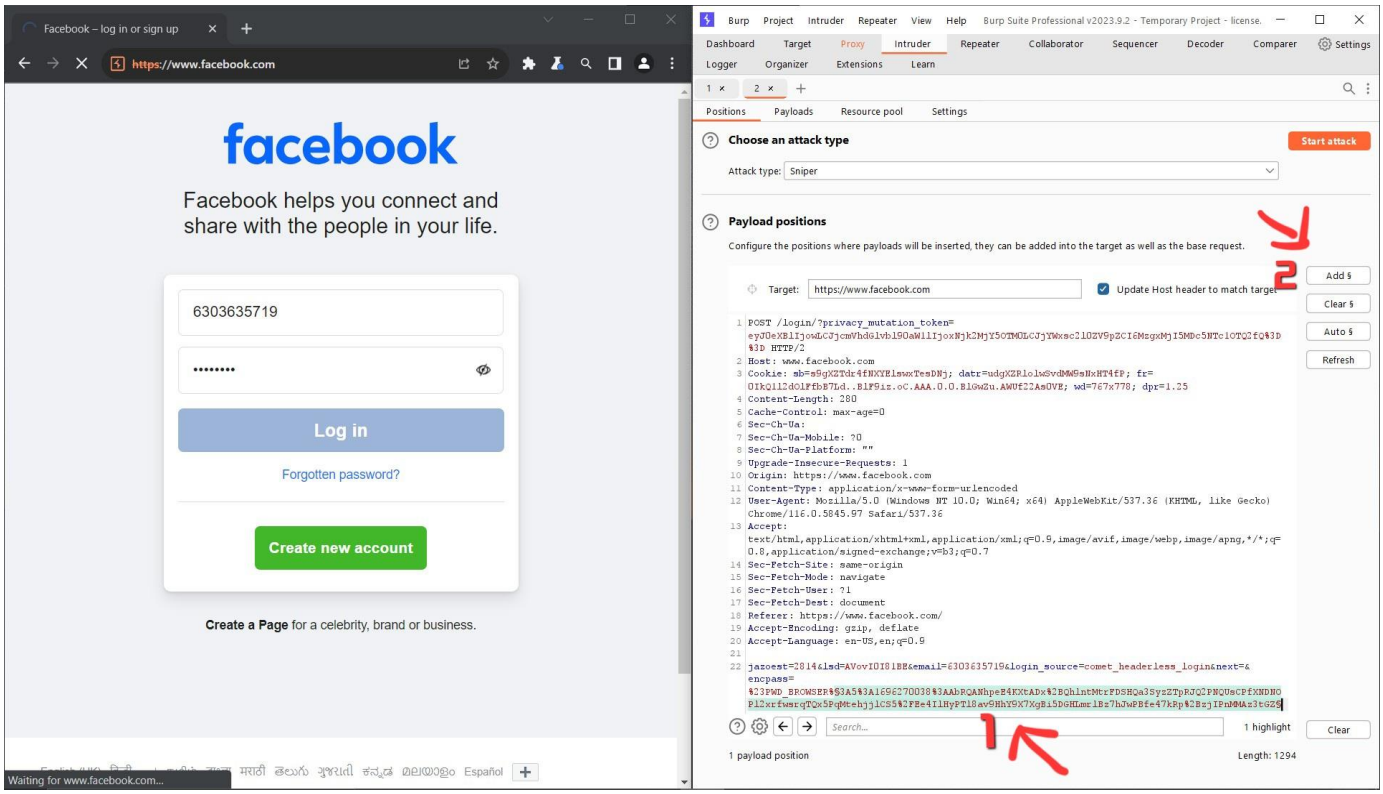
- Click on action and select "send to Repeater".



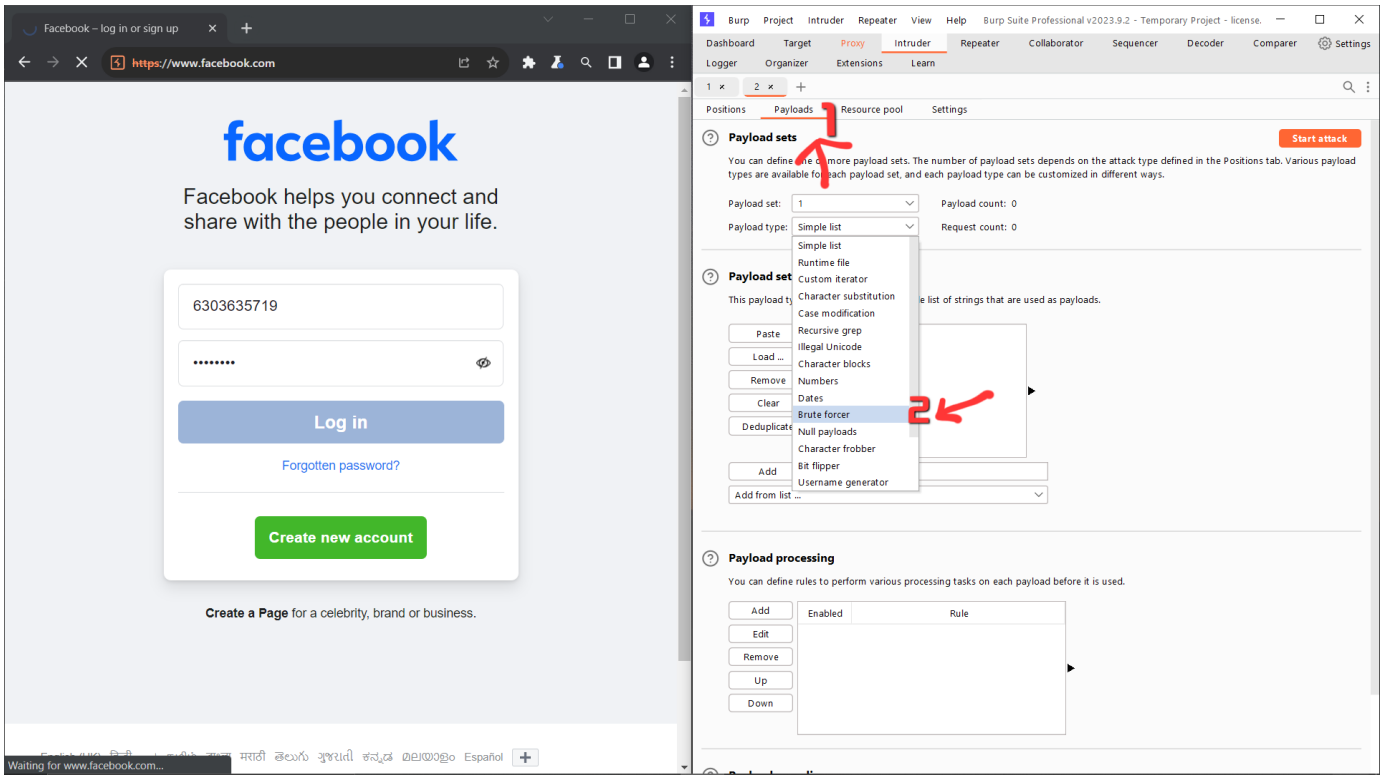
- Move to Repeater and click send button (1) and check whether there is “200 OK” is there in response or not (2)



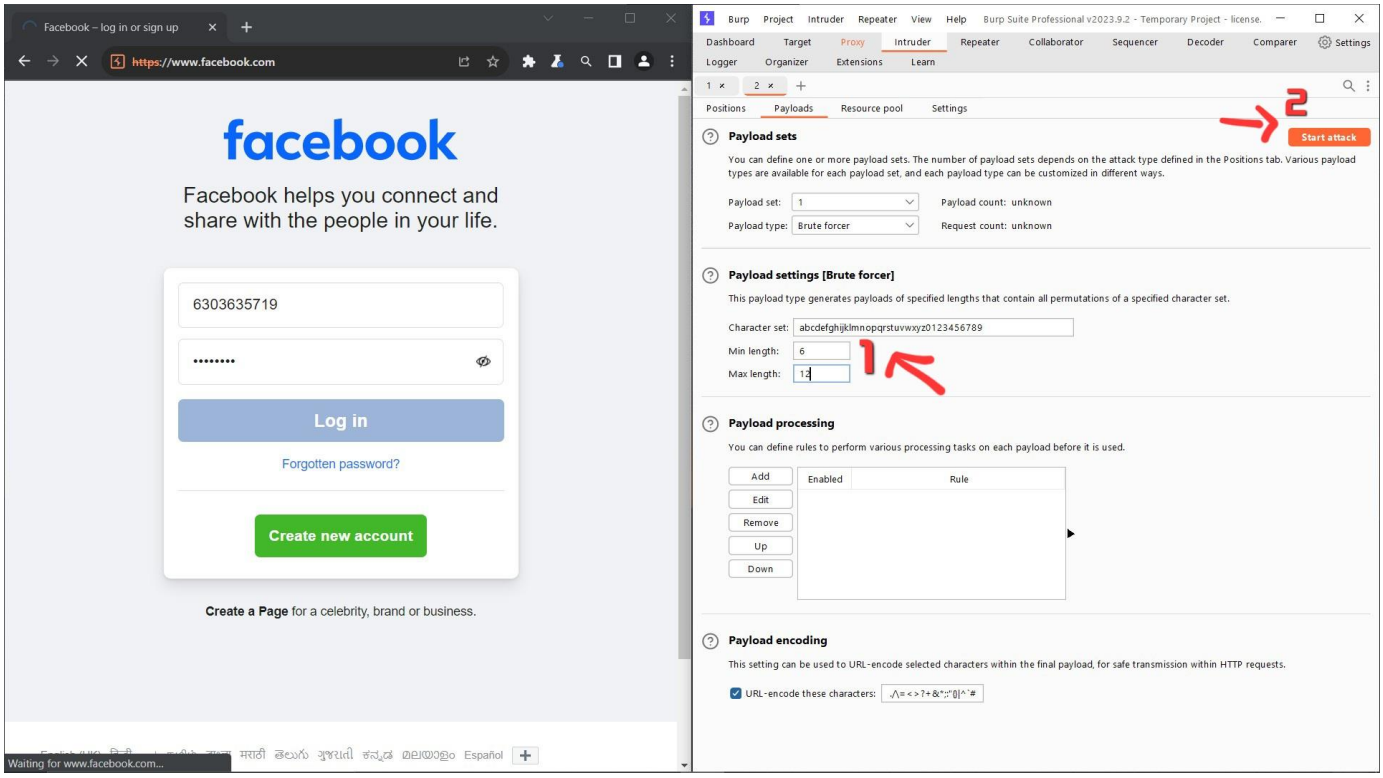
- If there is "200 OK" in response, move to Proxy and click on action and then select "Send to Intruder".



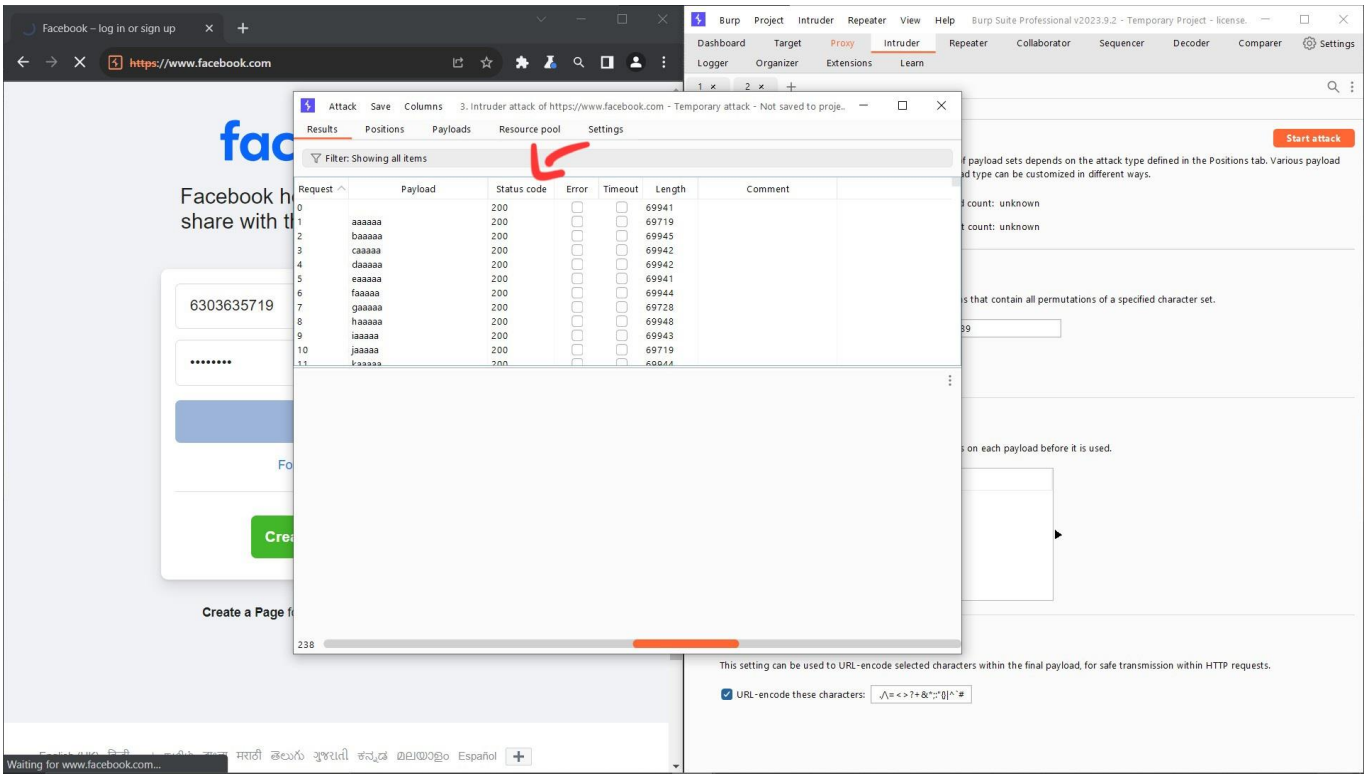
- Move to Intruder and select the encrypted format of password as shown above (1) and click on "ADD" (2).



➢ Move to Payloads (1) and select payload type as "Brute forcer" (2).



- Customize payload settings for searching password according to website (1) like alphabets in small letters and capital letters, numbers and special characters which might contain in passwords. Then click on "Start attack" (2).



- Then attack will get started and when it found password, we can see "302" in status code for that password and the length of that password will be higher than others and we can see true statement when we selected that password.
- This is the process of capturing the target's password or username or OTP.

Remediation

Option 1: Multi-Factor Authentication (MFA) - Implement MFA to enhance user verification.

Option 2: Strong Password Policies- Enforce robust password creation and management rules.

Option 3: Account Lockout- Set policies for temporary lockout after repeated failed login attempts.

❖ Vulnerable and Outdated Components – LOW

Description

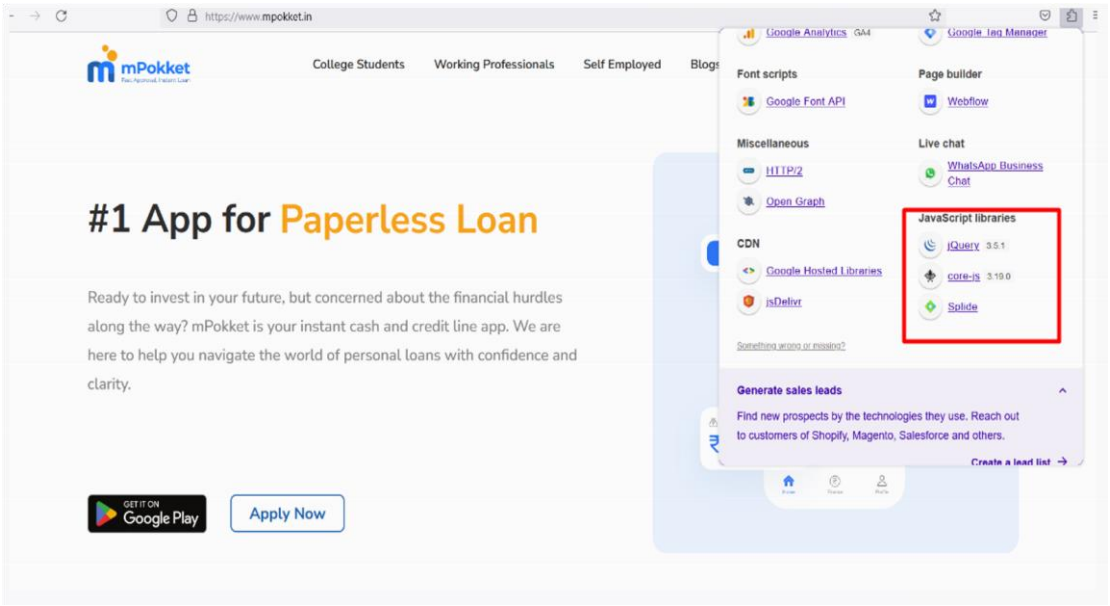
In today's ever-evolving digital landscape, the security of software and technology systems has become paramount. One of the critical challenges faced by developers, IT professionals, and organizations is the presence of vulnerable and outdated components within their software infrastructure. This issue, often referred to as "A06," represents a pressing concern for anyone committed to maintaining a robust and secure digital environment.

Vulnerable and outdated components are those hidden weak points in software applications that can be exploited by malicious actors, potentially leading to security breaches, data leaks, and significant business disruptions. To mitigate this risk, it is imperative to understand the nature of A06 and adopt effective strategies to identify, assess, and remediate these vulnerabilities.

Analysis

The assessment aimed to identify and analyze the presence of "Vulnerable and Outdated Components" within the mpokket website's technology stack. The following methodology was employed.

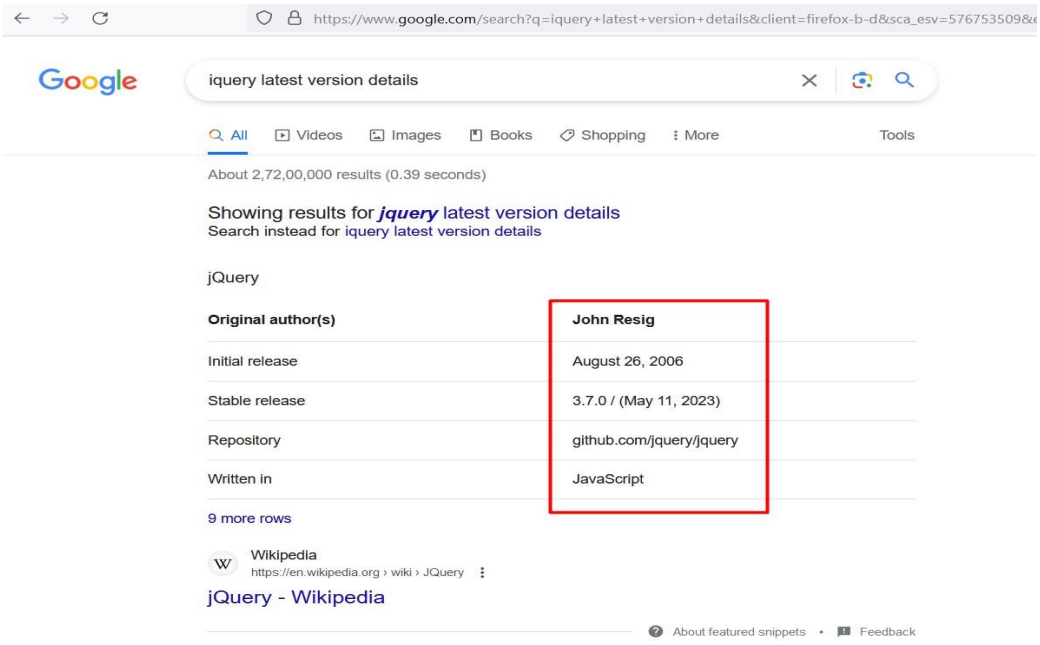
Component Identification:
To begin, an examination of the website's technology stack was conducted using the Wappalyzer tool. This tool provided insights into the various technologies and libraries utilized by the website, including JavaScript libraries.



Wappalyzer analysis reveals website technologies, forming the basis for identifying outdated components in subsequent steps.

VERSION ANALYSIS

The specific versions of JavaScript libraries were scrutinized to determine if any outdated or vulnerable components were in use. Particular attention was given to the jQuery library, with the discovery of version 3.5.1 raising immediate security concerns.



Verifying jQuery Version - The screenshot documents the process of checking the latest jQuery version on the official jQuery website, aiding in the assessment of outdated components and potential security risks.

Remediation

Option 1: Regular Updates- Keep all software components, libraries, and dependencies up to date.

Option 2: Vulnerability Scanning- Use automated tools to identify known vulnerabilities.

Option 3: Prompt Patching- Apply patches and updates swiftly to address vulnerabilities.

❖ Security Misconfiguration – MEDIUM

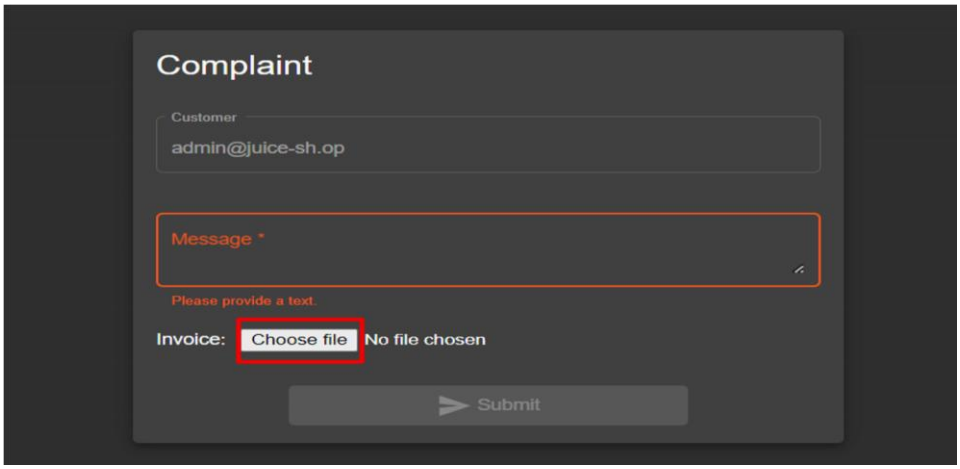
Description

This report focuses on a critical aspect of web security, namely "Security Misconfiguration," as observed in the file upload functionality of the OWASP Juice Shop website. Security misconfigurations are among the most common and dangerous vulnerabilities in web applications, often stemming from oversight or errors in the configuration settings. In this assessment, we delve into an instance where misconfiguration enabled the upload of unauthorized file types, specifically XML, which deviated from the intended accepted formats of zip and pdf. The objective of this assessment is to shed light on the security misconfiguration in the file upload functionality, outlining its implications for the OWASP Juice Shop website's security and integrity. By doing so, we aim to underline the critical importance of robust security configurations and recommend measures to rectify and prevent such vulnerabilities effectively.

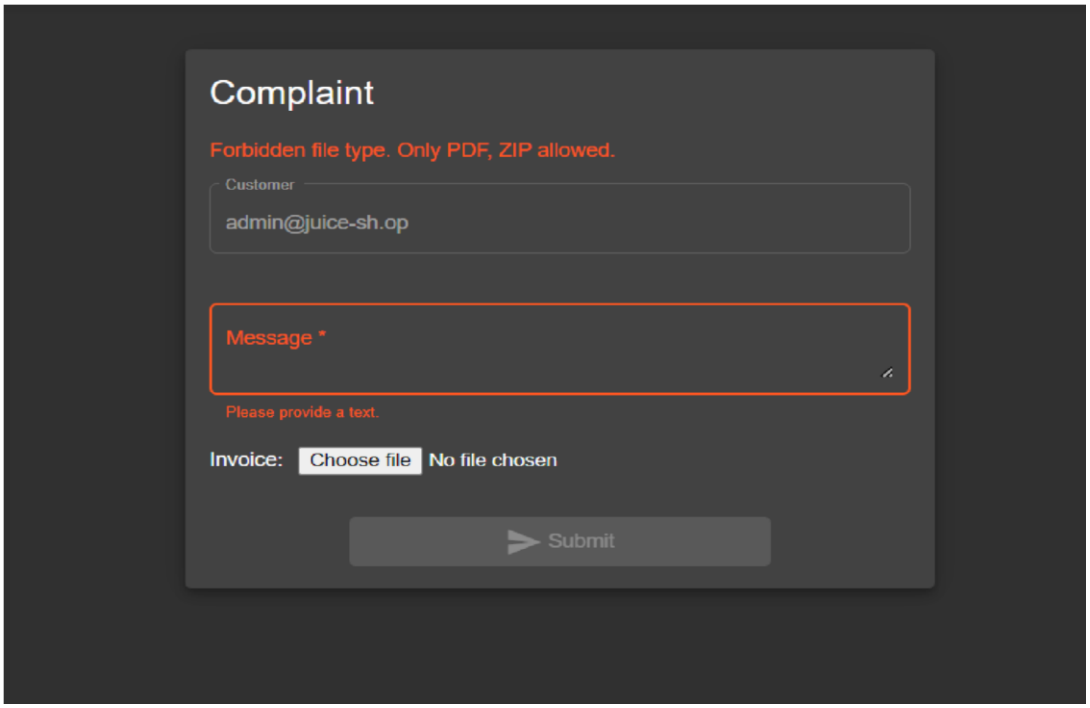
Analysis

STEP 1:

The identification of the "Security Misconfiguration" vulnerability commenced with a systematic review of the OWASP Juice Shop website's functionality. Our team initiated this process by examining the file upload feature, which was intended to accept only zip and pdf file formats.

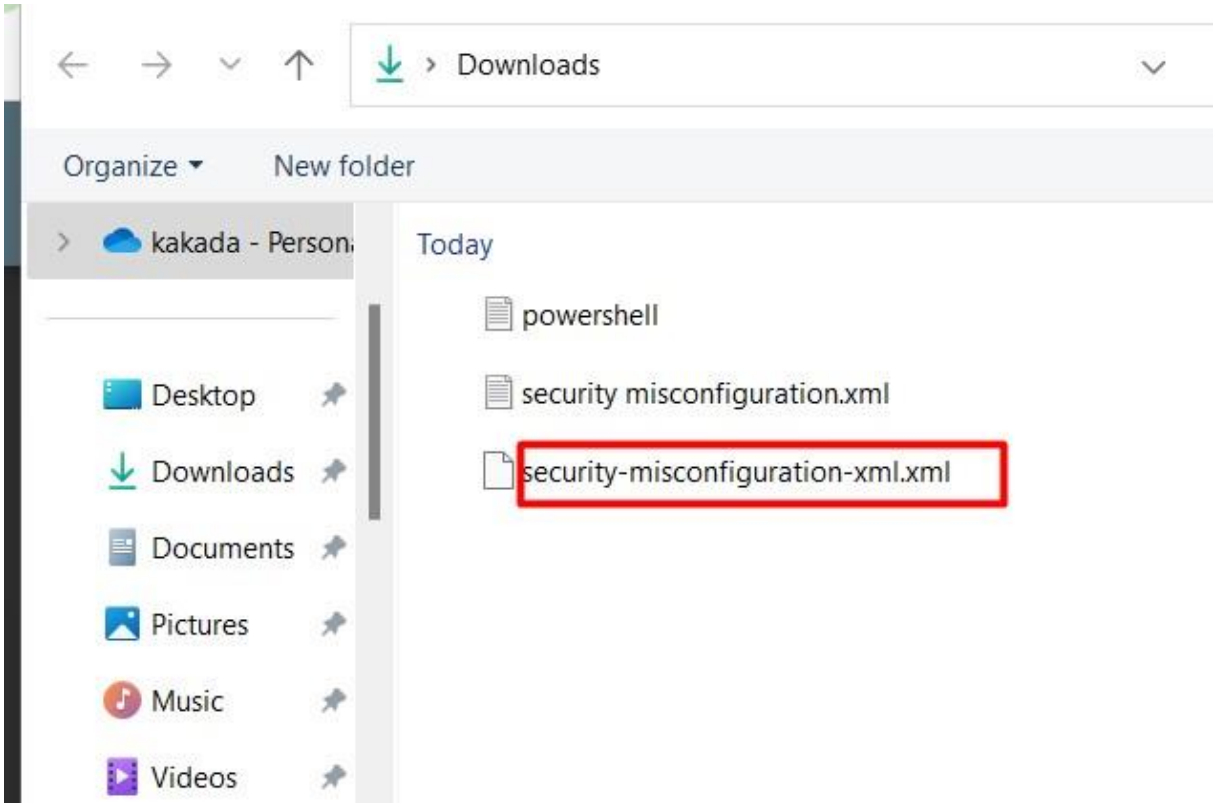


STEP 2:



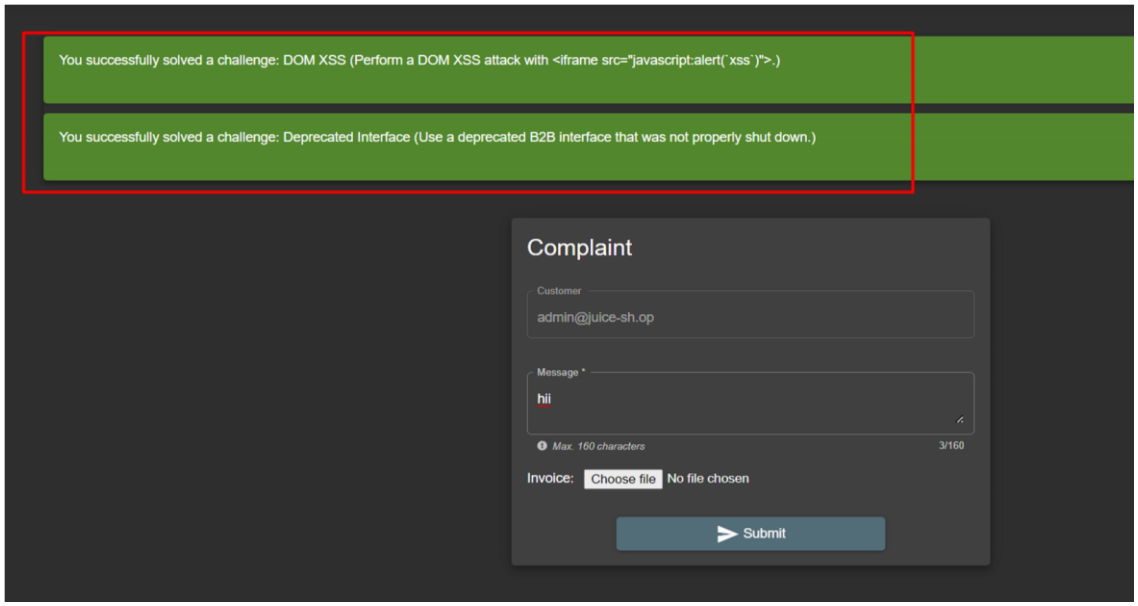
Here it shows it's allowed only pdf and zip file.

STEP 3:



Here we convert this file into an xml file. Now we have to find out if this website is taking that file or not.

STEP 4:



RESULT: This screenshot captures the moment when an unauthorized XML file was successfully uploaded through the file upload functionality, highlighting the security misconfiguration in the OWASP Juice Shop website.

Remediation

Option 1: Automated Scanning- Employ automated security scanning tools to identify and address misconfigurations.

Option 2: Least Privilege- Apply the principle of least privilege to limit access and permissions.

Option 3: Regular Auditing- Conduct routine security audits to check for misconfigurations.

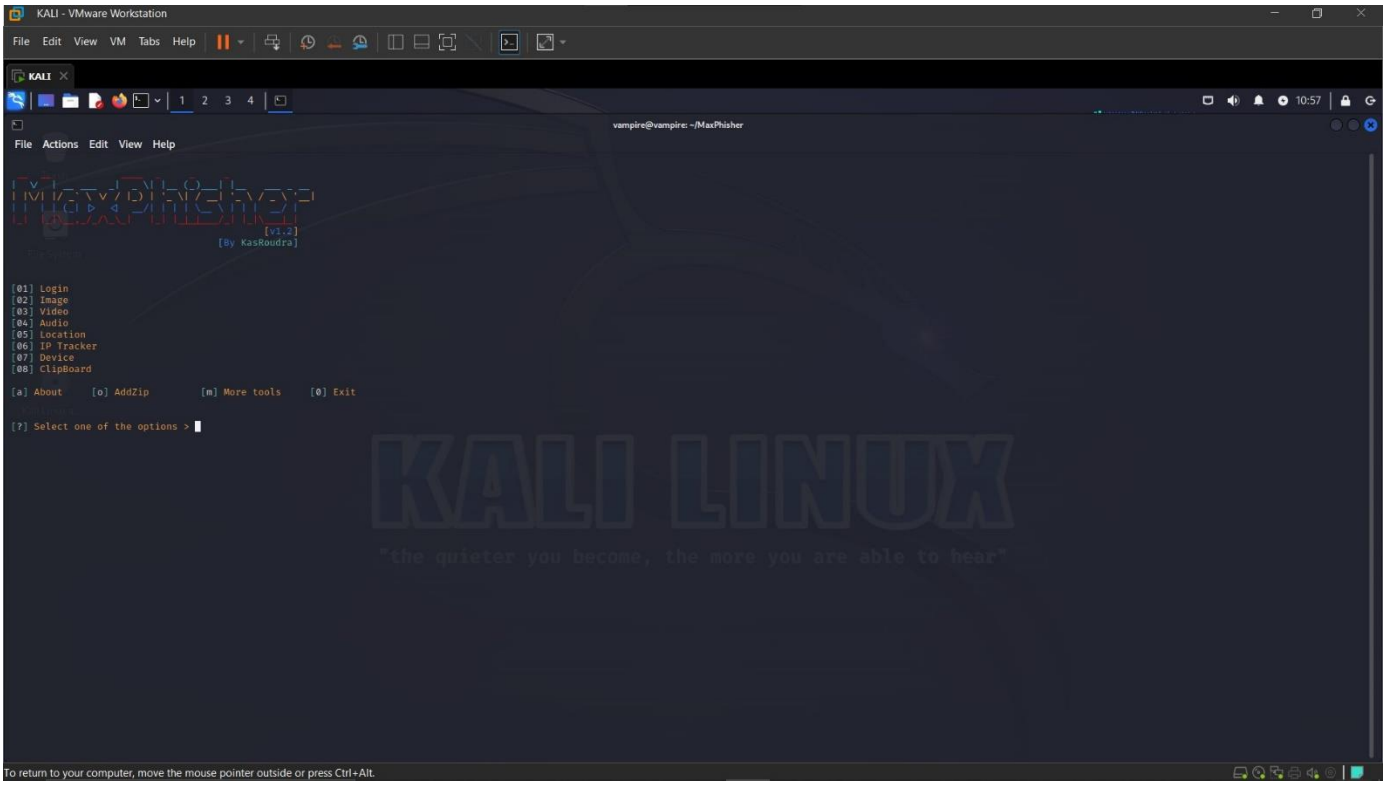
❖ Phishing Attack –HIGH

Description

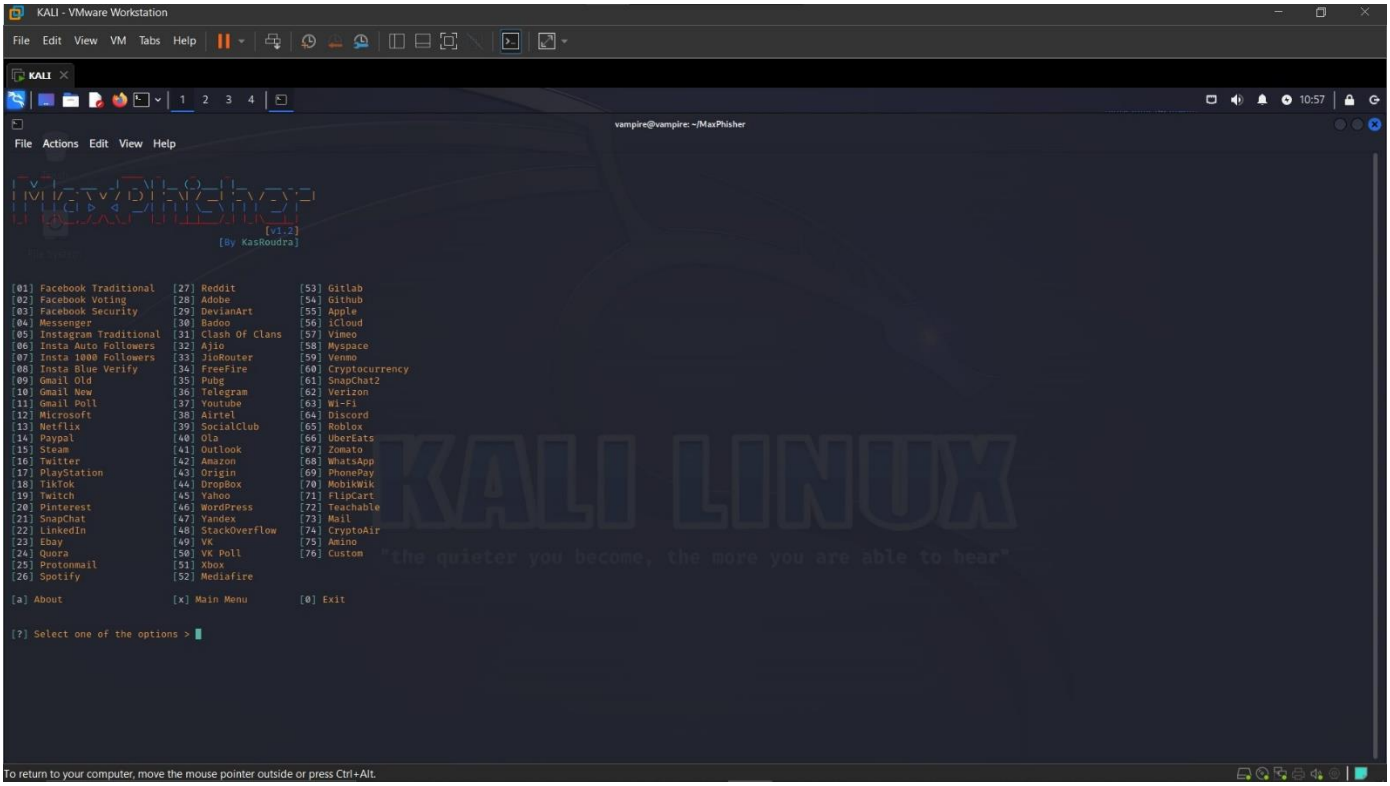
Phishing attacks are a prevalent and evolving cyber threat that jeopardize individuals and organizations' security and privacy. In these attacks, cybercriminals disguise themselves as trustworthy entities to deceive unsuspecting victims into revealing sensitive information or performing harmful actions. Phishing attacks typically occur through email, social engineering, or fraudulent websites. This poses a significant risk to personal and corporate data, financial resources, and even reputation. Therefore, it is crucial to understand these attacks and implement effective mitigation strategies to safeguard against them.

Analysis

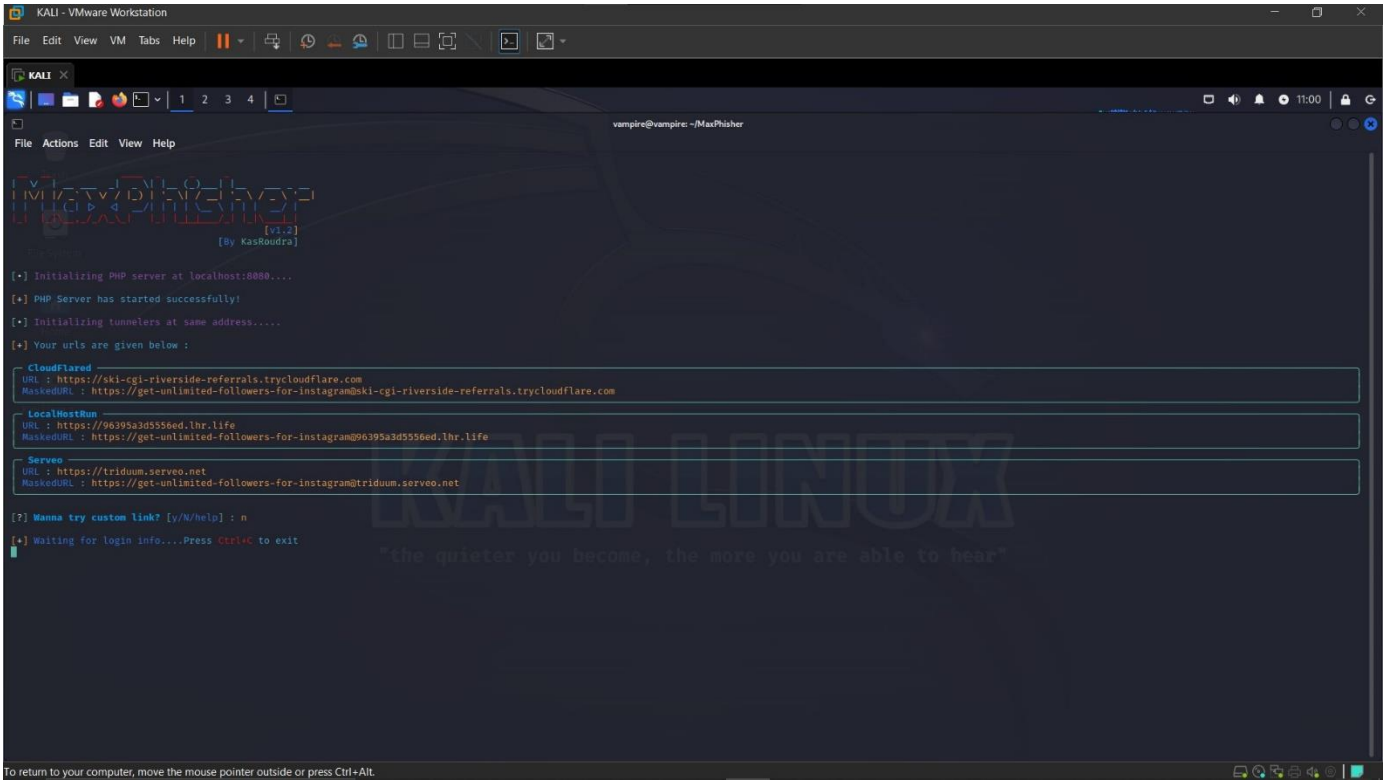
- Here we are going to use MaxPhisher tool for phishing.
- Clone the MaxPhisher tool from github in Linux terminal.
- Select Instagram login in MaxPhisher to perform phishing attack.



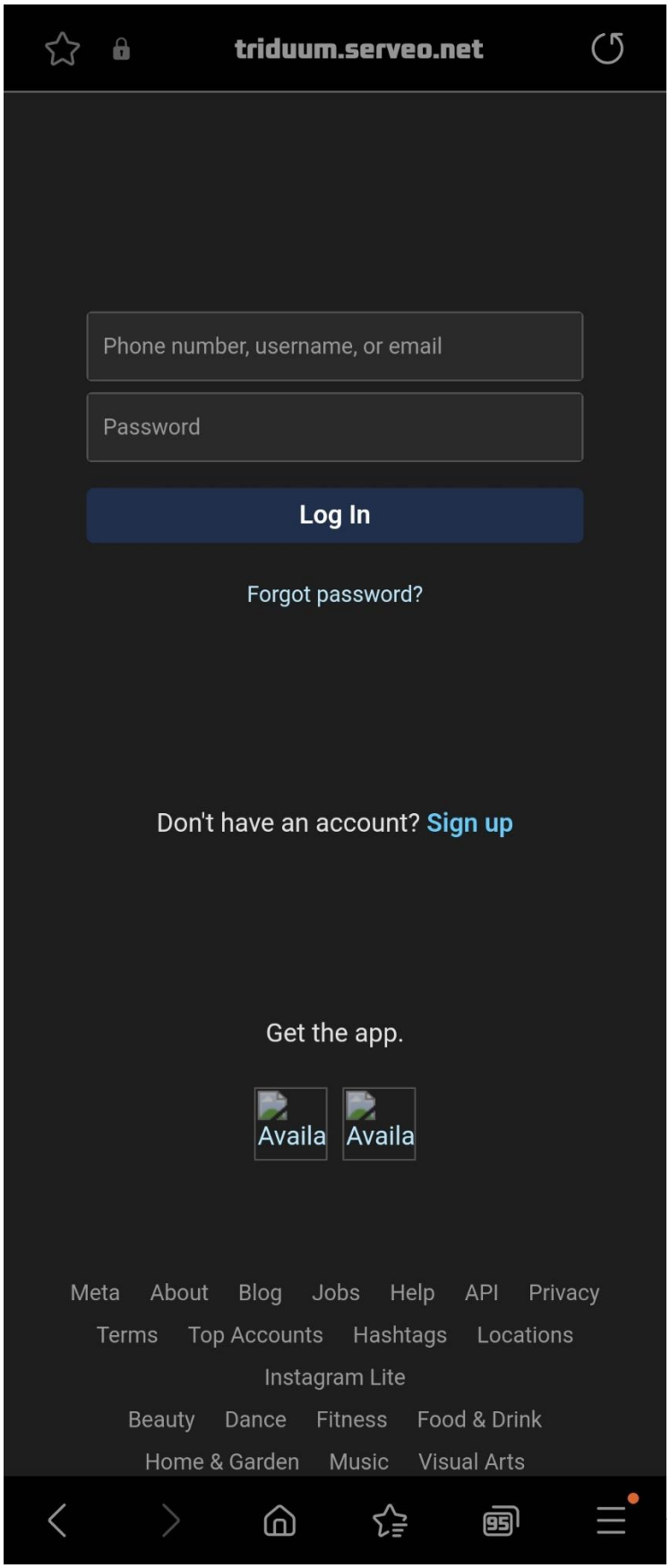
- Once it started running MaxPhisher, enter "01" to use login services of this tool.

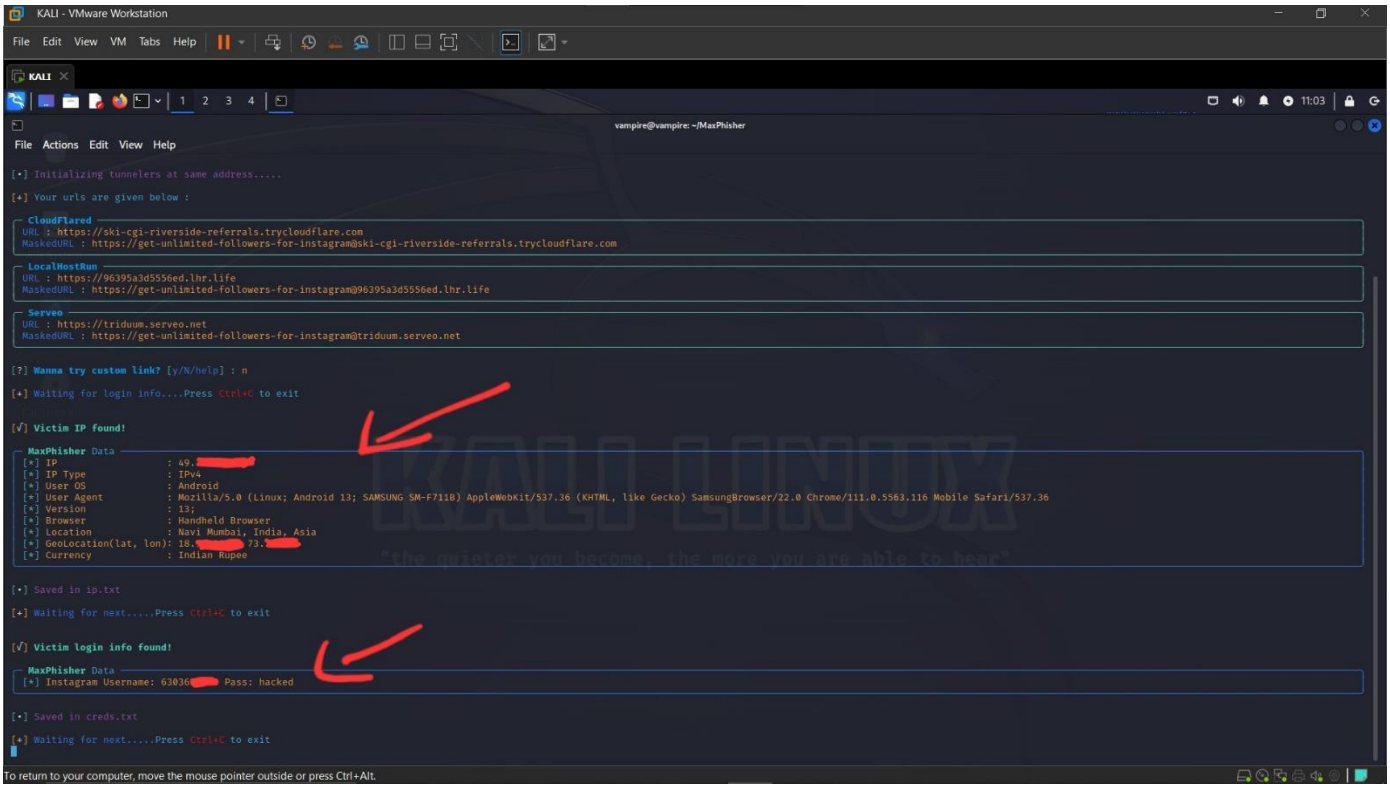


- For instagram login, enter “05” and click enter.



- Then it will generate the links, copy any one link and share to the target.
- It will generate fake instagram login page where target believes its original and enters the credentials.


triduum.serveo.net
Phone number, username, or email
Password
Log In
Forgot password?
Don't have an account? Sign up
Get the app.
Availa
Availa
Meta About Blog Jobs Help API Privacy
Terms Top Accounts Hashtags Locations
Instagram Lite
Beauty Dance Fitness Food & Drink
Home & Garden Music Visual Arts



- Here we captured username and password of target's account when target tried to login through the link shared.

Remediation

Option 1: Education and Awareness- The first line of defense against phishing attacks is education and awareness. Training employees and individuals to recognize phishing attempts is critical. They should be able to identify suspicious email addresses, subject lines, and content. Regular security awareness programs and simulated phishing exercises can help reinforce these skills.

Option 2: Multi-Factor Authentication (MFA) - Implementing MFA adds an extra layer of security by requiring users to provide multiple forms of identification before accessing an account. Even if attackers obtain login credentials, MFA can prevent unauthorized access.

Option 3: Regular Software Updates- Keep operating systems and software up to date to patch vulnerabilities that cybercriminals might exploit. Phishing attacks can target unpatched systems.

Option 4: HTTPS and Website Validation- Always verify the legitimacy of websites by checking for the "https" in the URL, indicating a secure connection. Look for visual cues, such as padlock icons, and avoid clicking on links from suspicious sources.

Conclusion

The black box penetration test conducted on the 24Chakra Organization's network security revealed significant vulnerabilities, including both critical and lower-severity issues. The organization's security posture was found to be inadequate, largely due to human-related errors, such as patch management issues and non-compliance with industry best practices.

The key takeaways from this vulnerability assessment can be summarized as follows:

- Risk Awareness: The assessment has provided a clear understanding of the existing vulnerabilities and associated risks within the 24 Chakra.
- Prioritization of Action: The vulnerabilities have been categorized based on their severity, enabling us to prioritize and address the most critical issues promptly.
- Security Controls: Mitigations have been provided to strengthen security controls and practices.

In conclusion, www.24chakra.com demonstrates a commitment to security by effectively mitigating many common web application vulnerabilities. However, there are opportunities for improvement to further enhance the site's security posture. By addressing the identified issues and implementing the recommended mitigations, the website can maintain and strengthen its resilience against potential threats and provide a secure and reliable user experience. Ongoing security monitoring and regular assessments are essential to continually improve web application security and maintain user trust.

BUG BOUNTY REPORT

Allows Disposable Email Addresses

ID	5969e65a-ddfd-4ba3-afe3-af19f332ab04
Submitted	03 Nov 2023 11:44:58 UTC
Target	Hubspot.com
Target	Others
VRT	Insufficient security configuration > weak registration Implementation > Allows Disposable email Addresses
Priority	P5
Bug URL	https://app.hubspot.com/signup-hubspot/cr m?hubs_signup-cta=login-signup-cta&hubs _signup-url=app.hubspot.com%2Flogin&uuid=971b038c-dac5-428a-928a-a26bb7312f9 e&step=landing_page

Description

Overview of the Vulnerability

When the registration implementation for an application is weak, it diminishes the integrity of the overall authentication process. The application allows users to submit a disposable or alias email address to register an account. An attacker can abuse this weakness to bulk register fake user profiles and use them to launch spam campaigns.

Business Impact

Having a weak registration implementation can result in reputational damage for the business through the impact to customers' trust as they could believe that the business doesn't take their account security seriously or trust that their data within will remain secure.

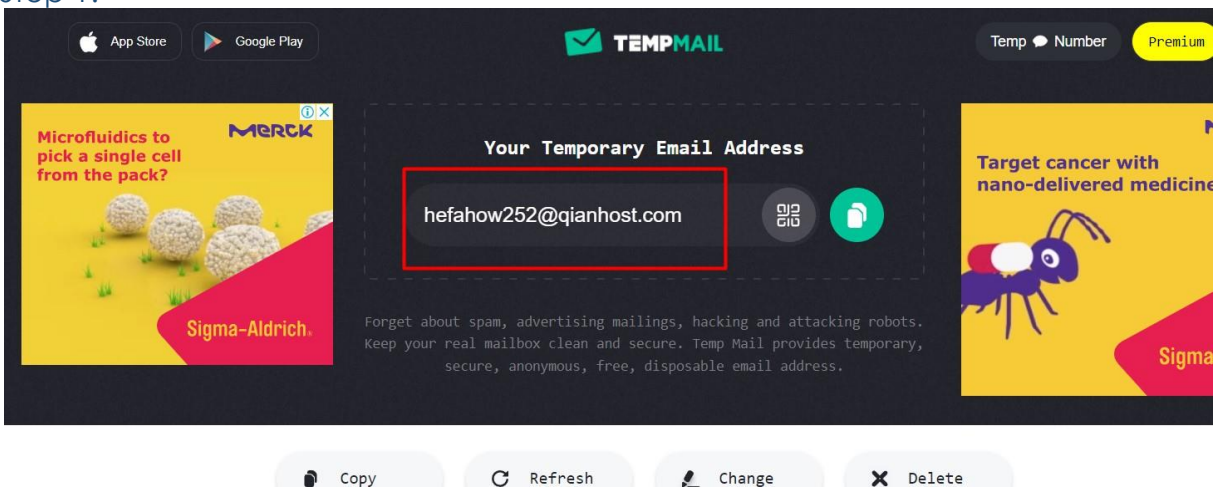
Steps to Reproduce

1. Use a browser to navigate to: https://app.hubspot.com/signup-hubspot/crm?hubs_signup-cta=login-signup-cta&hubs_signup-url=app.hubspot.com%2Flogin&uuid=971b038c-dac5-428a-928a-a26bb7312f9e&step=landing_page
2. Register an account using a disposable email service
3. Observe that the account is created

Proof of Concept (PoC)

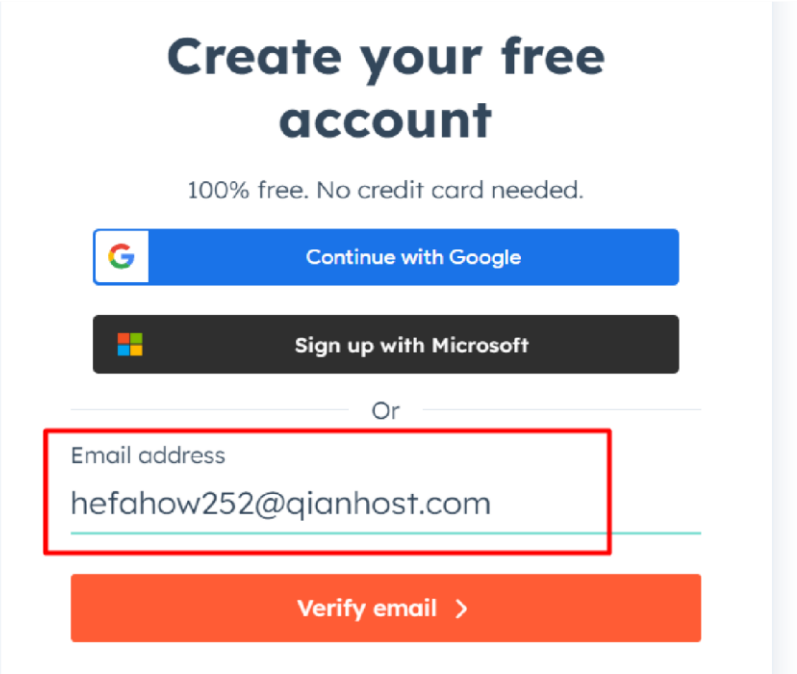
The following screenshot shows the weak registration implementation:

Step 1:



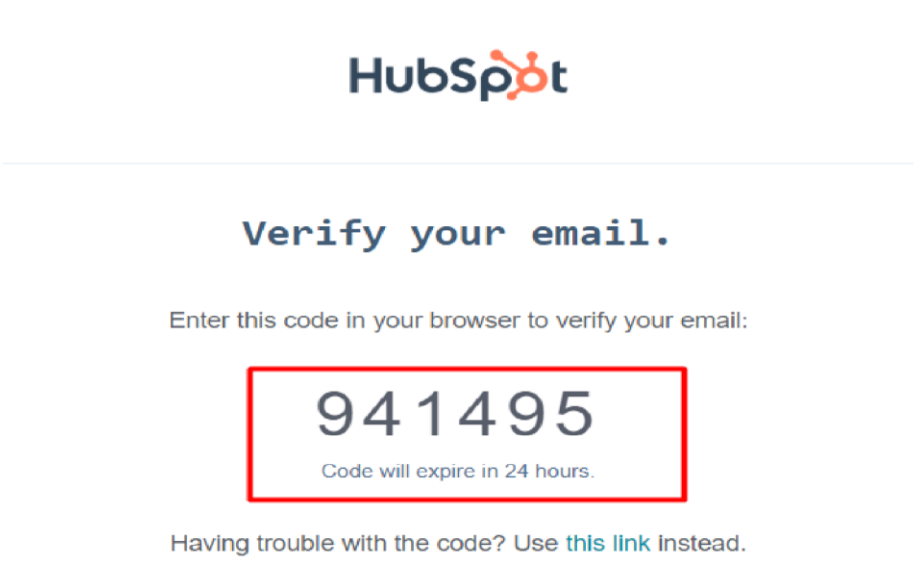
Capturing temporary mail id using Temp mail website.

Step 2:



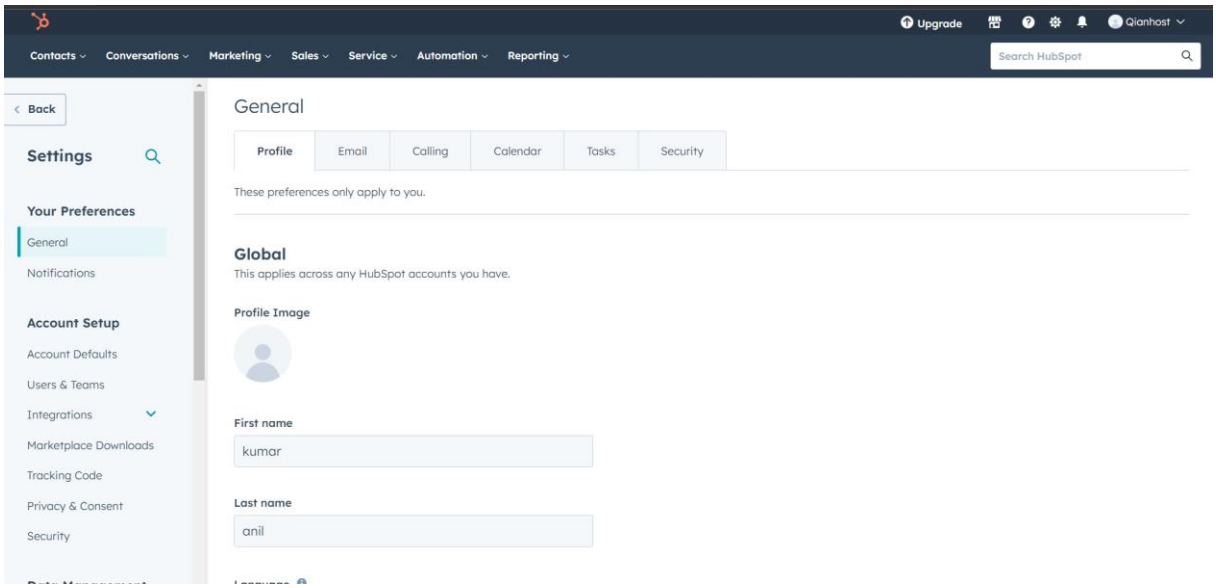
Entered that fake email credentials into the hubspot login panel for verification purposes.

Step 3:



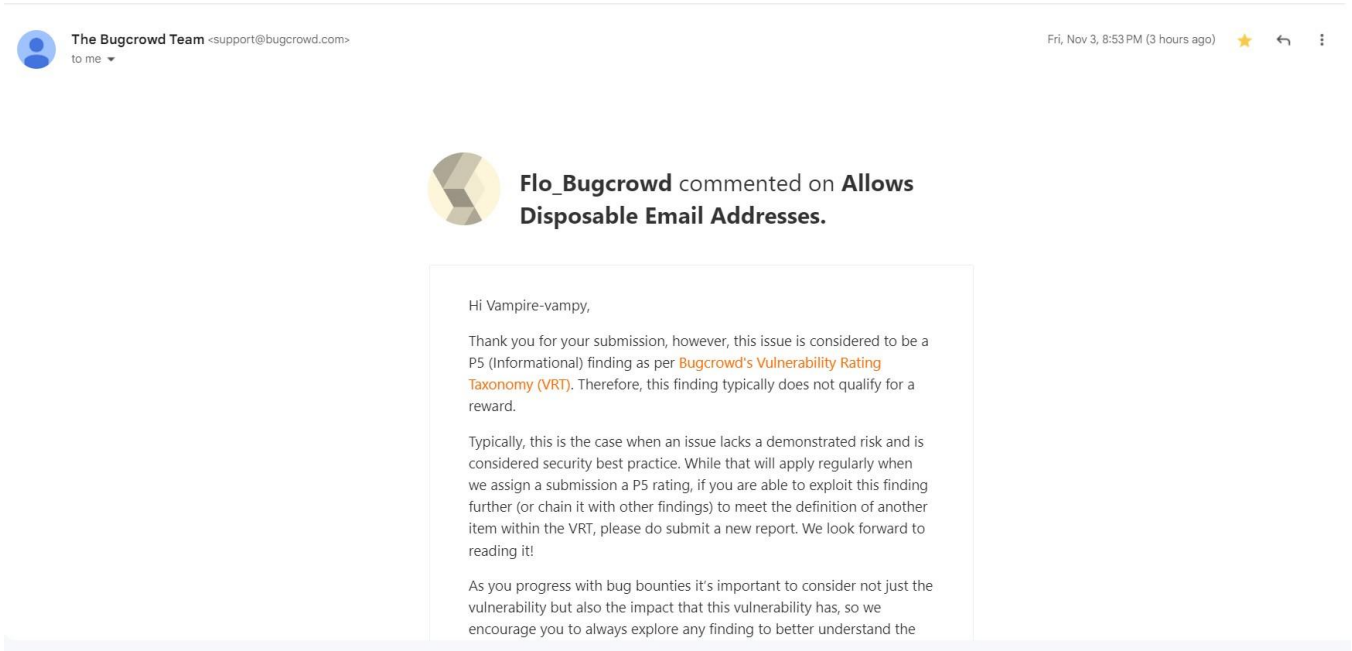
Through email Id I get OTP verification code for verification purposes.

Step 4:
Entered the verification code.



After entering the verification code. I successfully created account in Hubspot with the help of disposable mail.

RESPONSES: Getting a response from BUG CROWED for submitting a bug in that platform.



RESPONSE CODE: P5 (Informational)