Vulnerability Assessment and Penetration Testing Report

Table of Contents

Web Application Penetration Testing

1.1 Executive Summary

This is the result of the Vulnerability Assessment and Penetration Testing (VAPT) done on **https://testphp.vulnweb.com.** The main purpose of this assessment was to detect security vulnerabilities, exploit them under controlled conditions, and suggest remediation. Automated scanning tools and manual exploitation mechanisms were both used in the process of testing. The results reveal different security threats that an attacker could potentially exploit.

1.2 Background

This exercise aimed to perform Penetration Testing of the Applications in scope to determine if they were vulnerable to attacks and exploitation. The test consisted of manual testing to detect and exploit vulnerabilities.

The assessment was conducted to identify the security vulnerabilities in the Application in scope and propose solutions for the project team to remediate the identified vulnerabilities to make the Application more secure.

The Security Assessment was performed by Arshlan between **2nd April 2025** to **3rd April 2025.**

1.3 Objectives

The objective of the tests performed was to:

- Identify the possible vulnerabilities and gaps in the web applications.
- Assess the gaps and vulnerabilities to determine the Risk associated.
- Suggest recommendations to overcome existing vulnerabilities and gaps.

This assessment report contains:

- **Technical details** of the vulnerabilities discovered with substantiation of the exploits.
- **Risk mitigation recommendations** that need to be implemented to ensure that the systems are secure from the risks arising due to the discovered vulnerabilities.

1.4 Scope of Assessment

The scope for this Security assessment includes, but is not limited to, the following tests:

- **Target Application**: https://testphp.vulnweb.com
- **Testing Approach**: Black-box penetration testing
- **Security Areas Covered:**
    - SQL Injection (SQLi)
    - Cross-Site Scripting (XSS)
    - Cross-Site Request Forgery (CSRF)
    - Broken Authentication

1.5 Out of Scope

The following areas were considered out of scope for this assessment:

- Attacks that could cause permanent damage or disrupt the application.
- Social engineering, phishing, or other human-targeted attacks.
- Exploitation of vulnerabilities that require administrative access.
- Any testing beyond the given target (testphp.vulnweb.com).

1.6 Tools Used

I used a combination of manual testing and commercial/open-source tools as part of its penetration testing methodology. This is accompanied by custom scripts to ensure optimum results.

The following tools are used for the testing.

- Burp Suite
- Zap
- Mozilla Firefox
- Kali Linux

1.7 Summary of Findings

The Web Vulnerability Assessment and Penetration Testing (VAPT) conducted for the target yielded important findings and insights. This summary provides an overview of the key results obtained during the assessment.

It was observed that the application was exposed to a total of 4 security vulnerabilities, categorized as 4 Medium severity vulnerability during the assessment period.

S.NO	Name	Severity	Risk Score	Status
1	SQL Injection (SQLi)	Medium	7.0	Unresolved
2	Cross-Site Scripting (XSS)	Medium	6.2	Unresolved
3	Broken Authentication	Medium	7.3	Unresolved
4	Cross-Site Request Forgery (CSRF)	Medium	6.5	Unresolved

Vulnerabilities Details

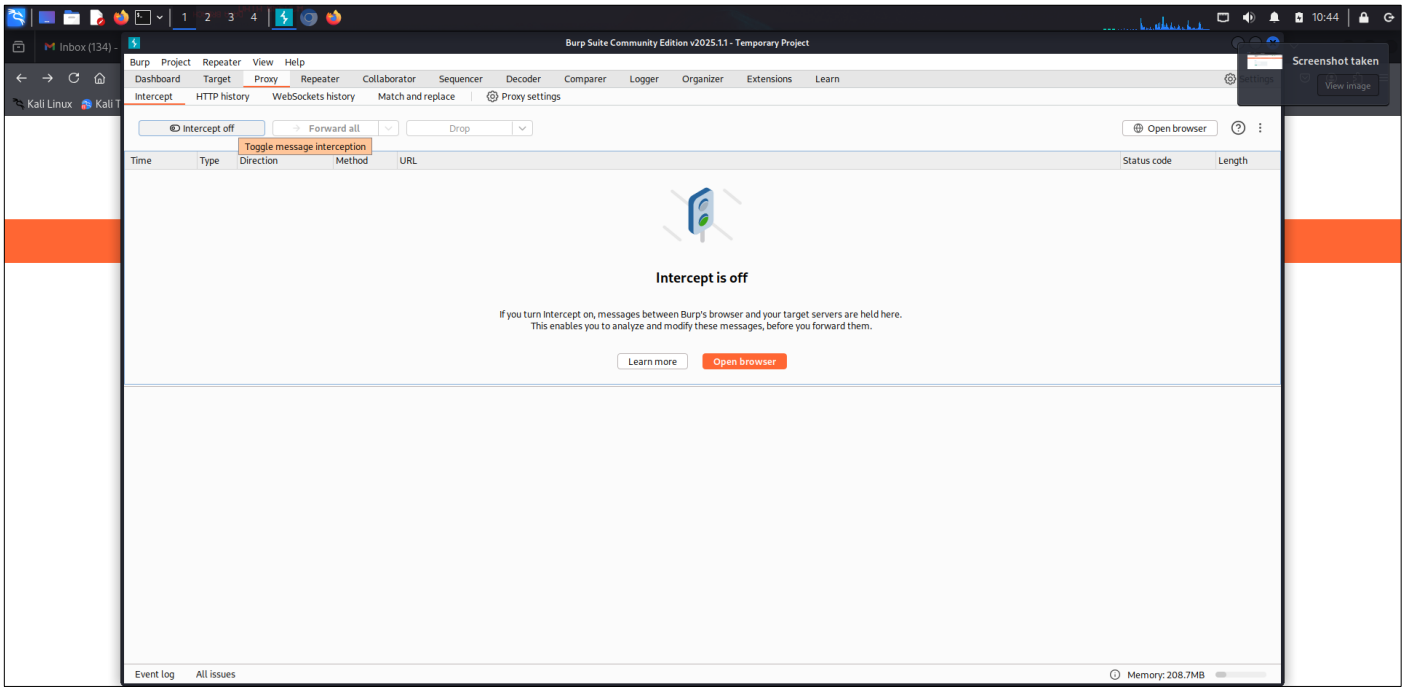
2.1 SQL Injection (SQLi) leads to Authentication Bypass

Parameter	Description
Severity	Medium
Impact	Medium
Risk Score	7.0
Affected URL	http://testphp.vulnweb.com/login.php
Threat	A successful SQL Injection attack can result in: 1. Authentication Bypass – Attackers can log in as any user, including administrators. 2. Data Theft – Confidential user credentials and sensitive data can be exposed. 3. Data Manipulation – Attackers can alter, delete, or insert data in the database. 4. Remote Code Execution – In some cases, SQLi can be leveraged for server takeover.

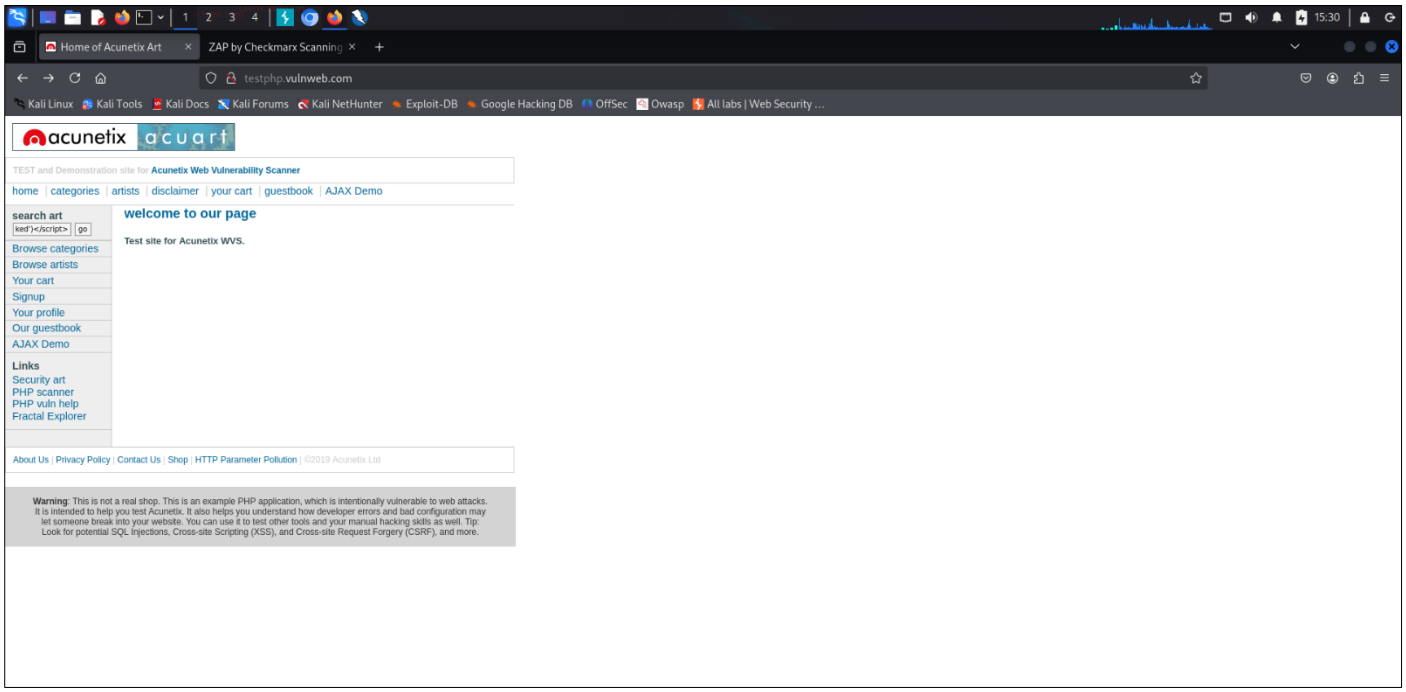
SQL Injection (SQLi) is a security vulnerability that allows attackers to manipulate the queries executed by a web application's database. This occurs when user-supplied input is improperly sanitized, allowing malicious SQL queries to be executed.

Steps To Reproduce

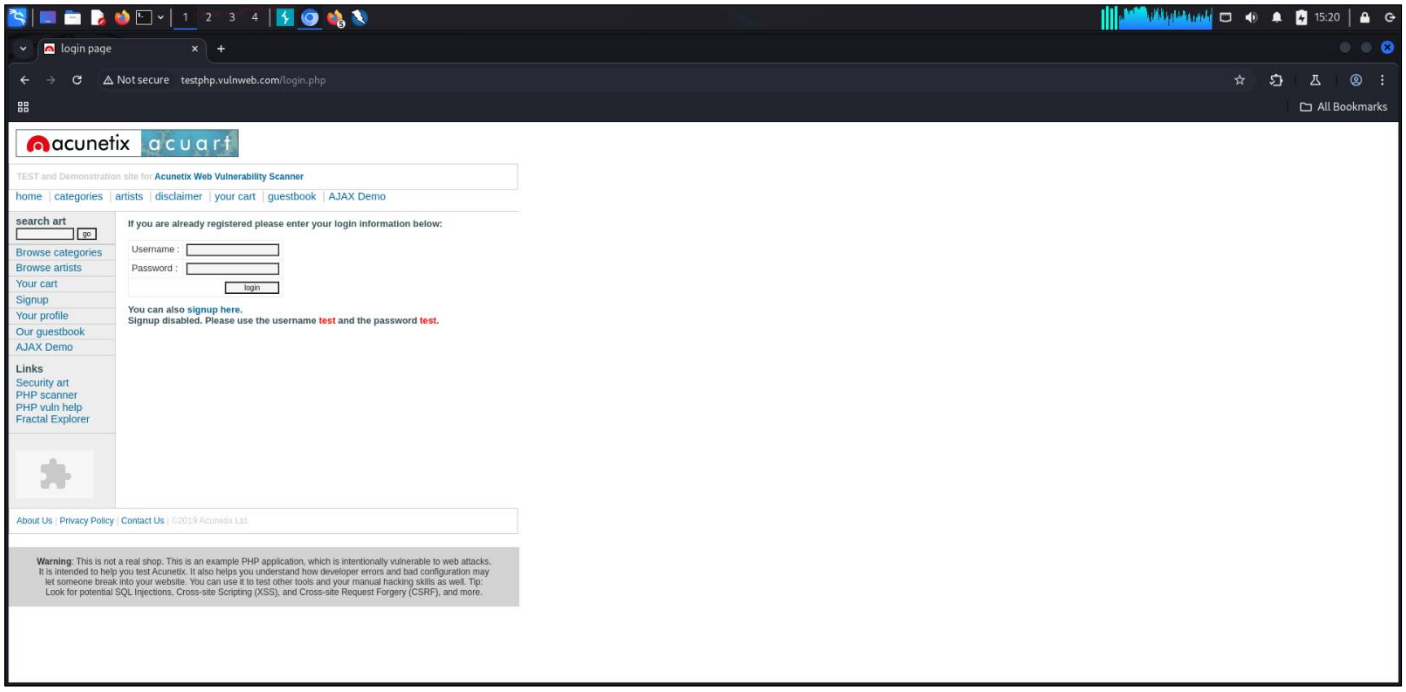
1. Open Burp Suite > Go to Proxy > Click on Open Browser.



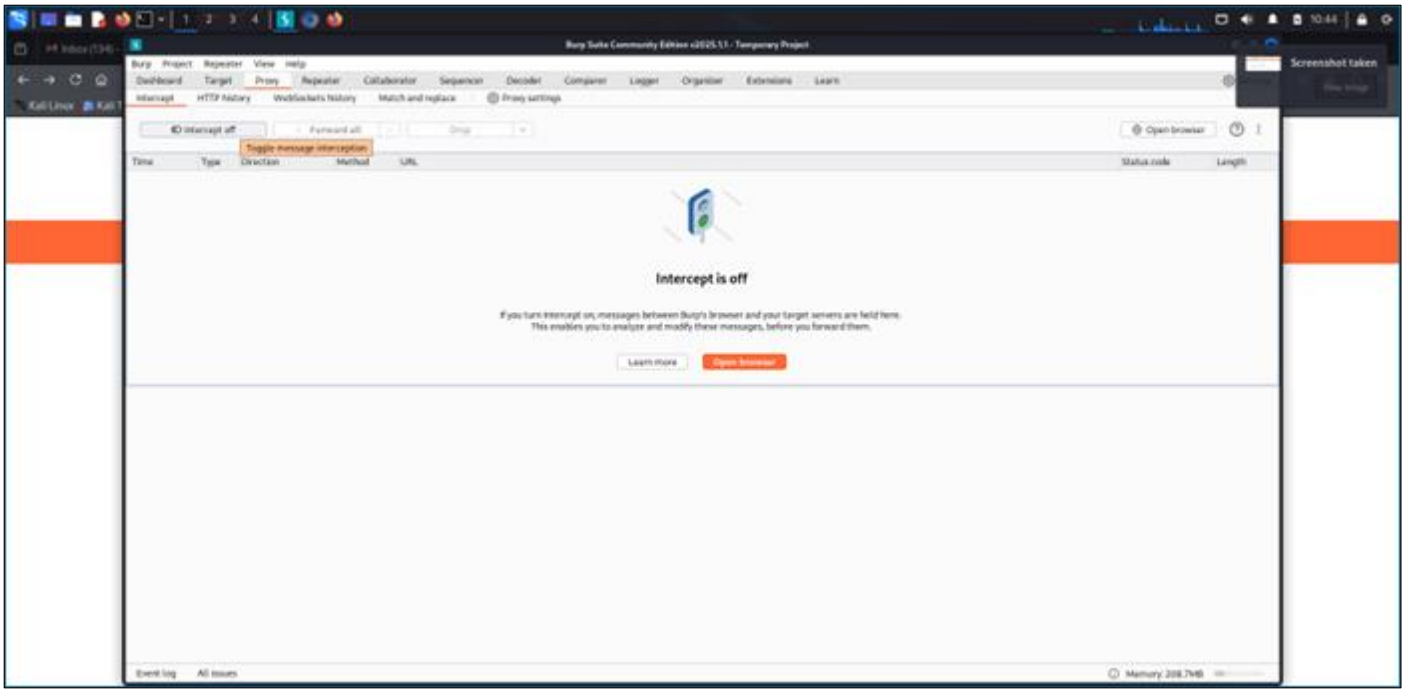
2. Access the given URL: http://testphp.vulnweb.com.



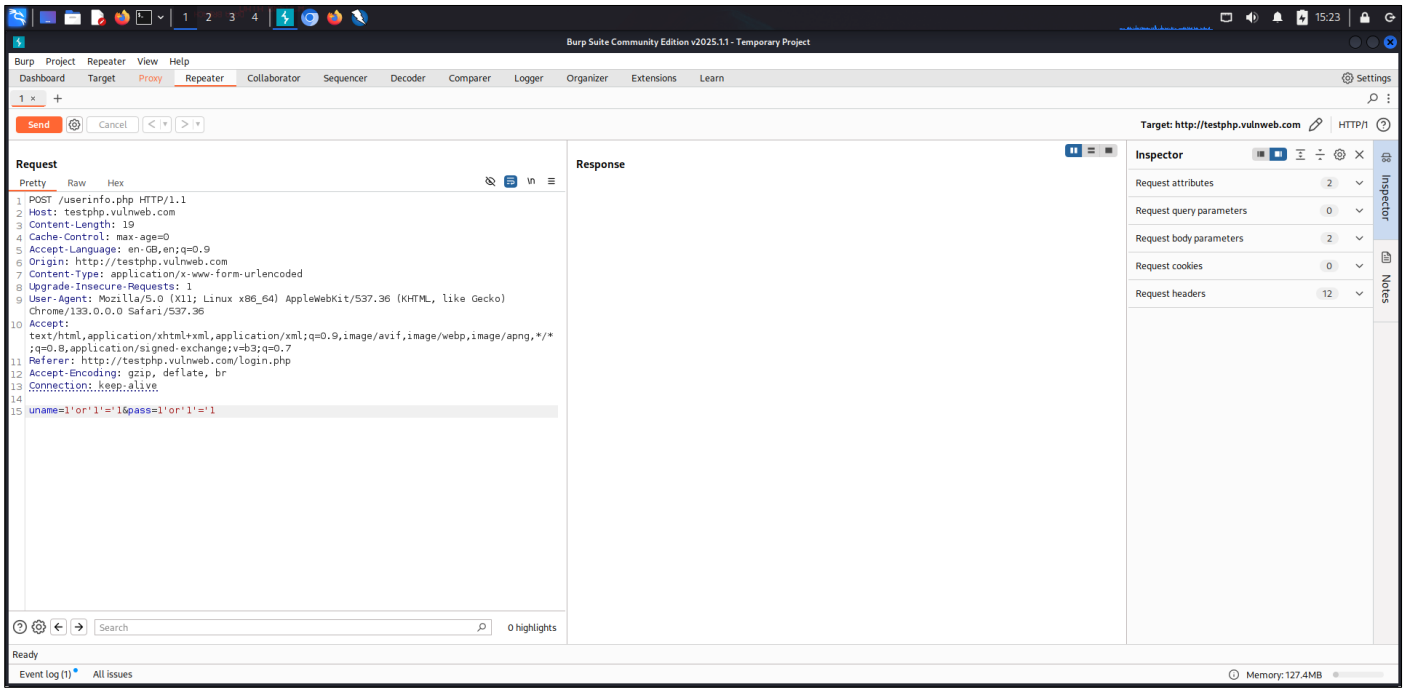
3. Click on Sign Up, enter any random username and password.



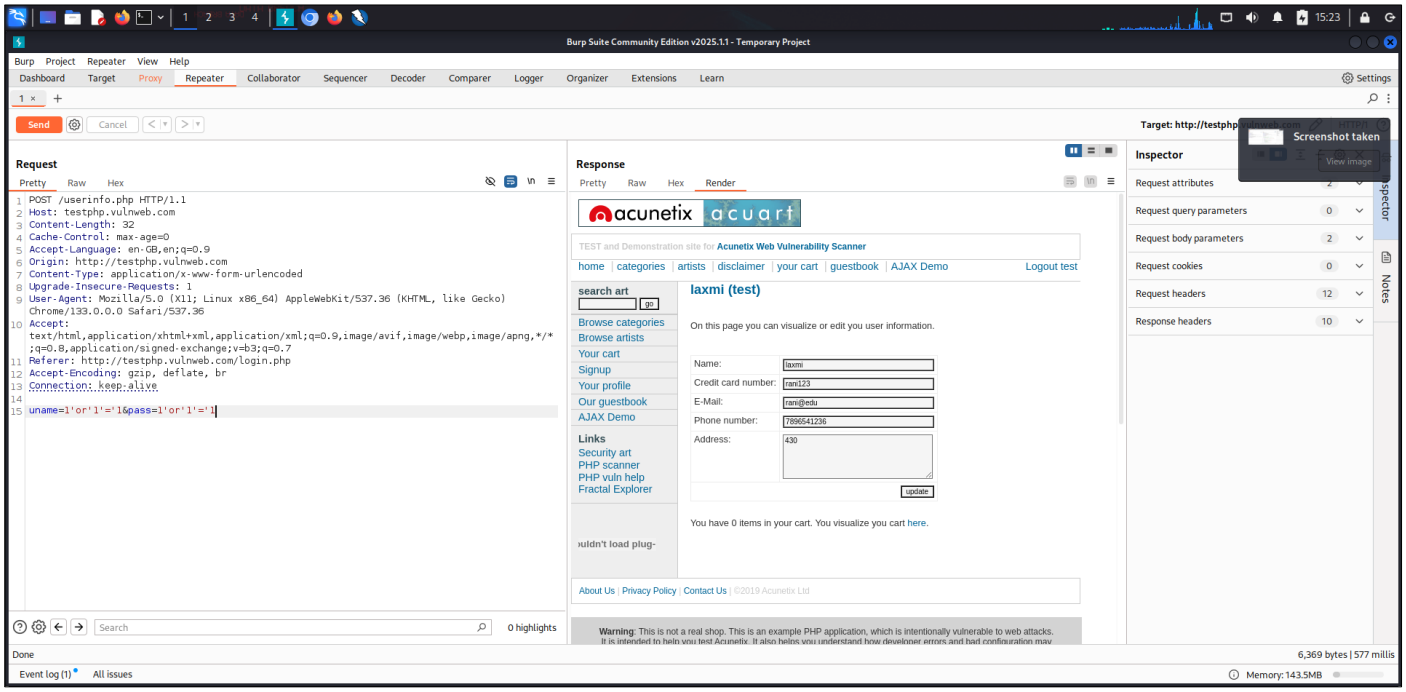
4. Before clicking Login, turn on Intercept in Burp Suite.



5. Submit the login form and capture the POST request > Right-click on the request and select Send to Repeater > In Repeater, modify the request payload to:
username=1'or'1'='1
password=1'or'1'='1



6. Click Send and check the response for 200 OK > render



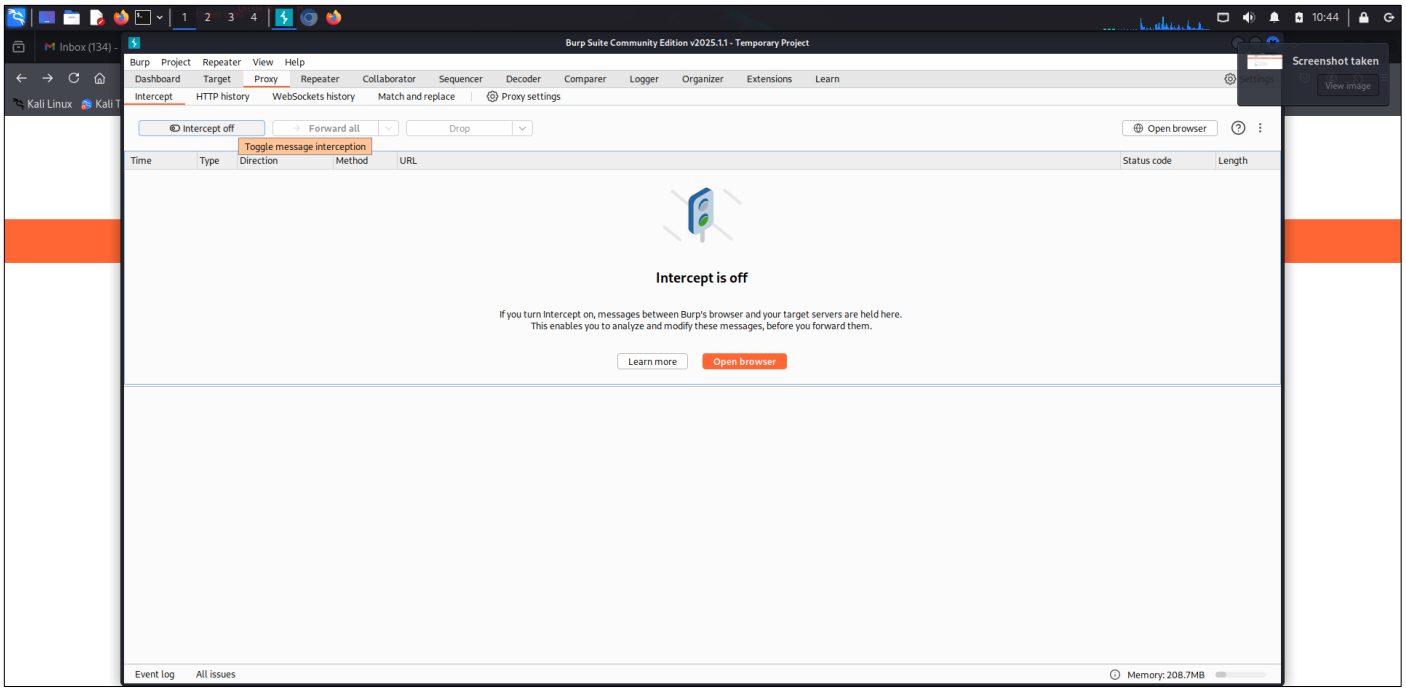
2.2 Cross-Site Scripting (XSS) on Search Input Box

Parameter	Description
Severity	Medium
Impact	Medium
Risk Score	6.2
Affected URL	http://testphp.vulnweb.com/search.php?test=query
Threat	A successful SQL Injection attack can result in: 1. Authentication Bypass – Attackers can log in as any user, including administrators. 2. Data Theft – Confidential user credentials and sensitive data can be exposed. 3. Data Manipulation – Attackers can alter, delete, or insert data in the database. 4. Remote Code Execution – In some cases, SQLi can be leveraged for server takeover.

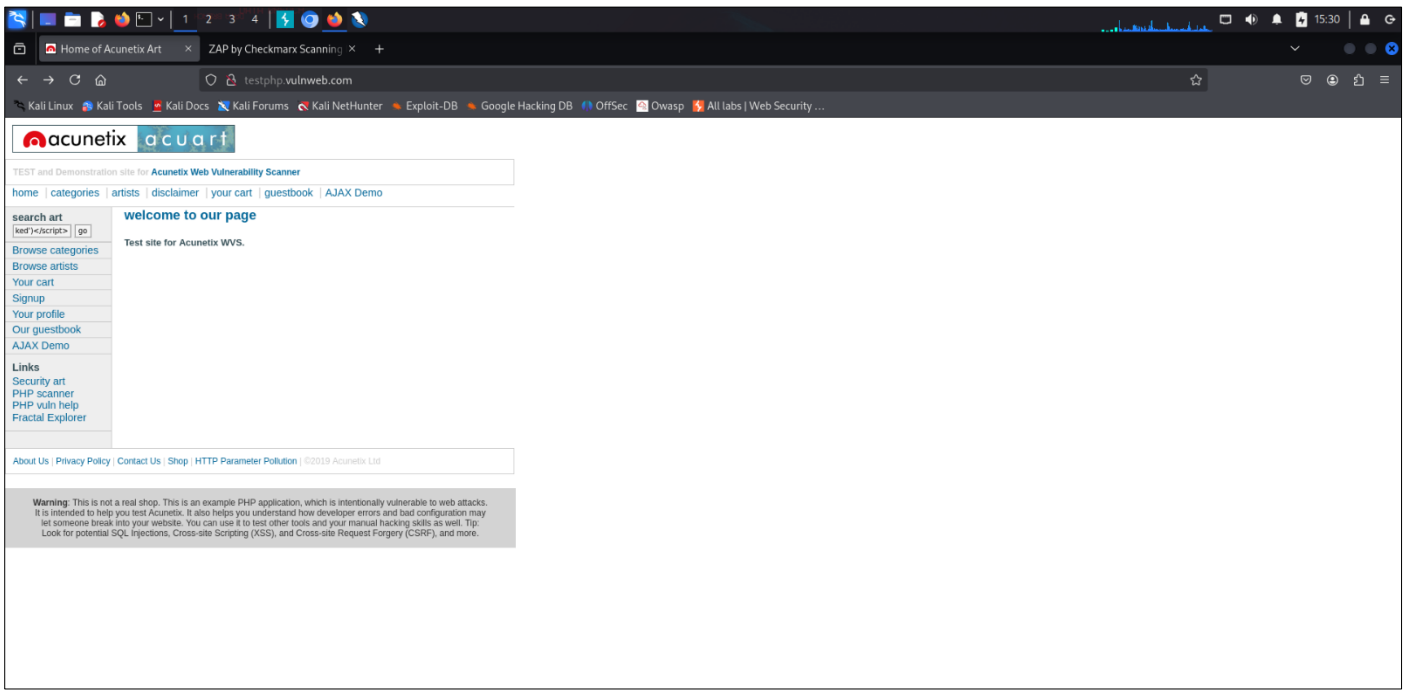
Cross-Site Scripting (XSS) is a vulnerability that allows attackers to inject malicious scripts into web pages viewed by other users. When the web application improperly handles user input, attackers can execute JavaScript in the victim's browser.

Steps To Reproduce

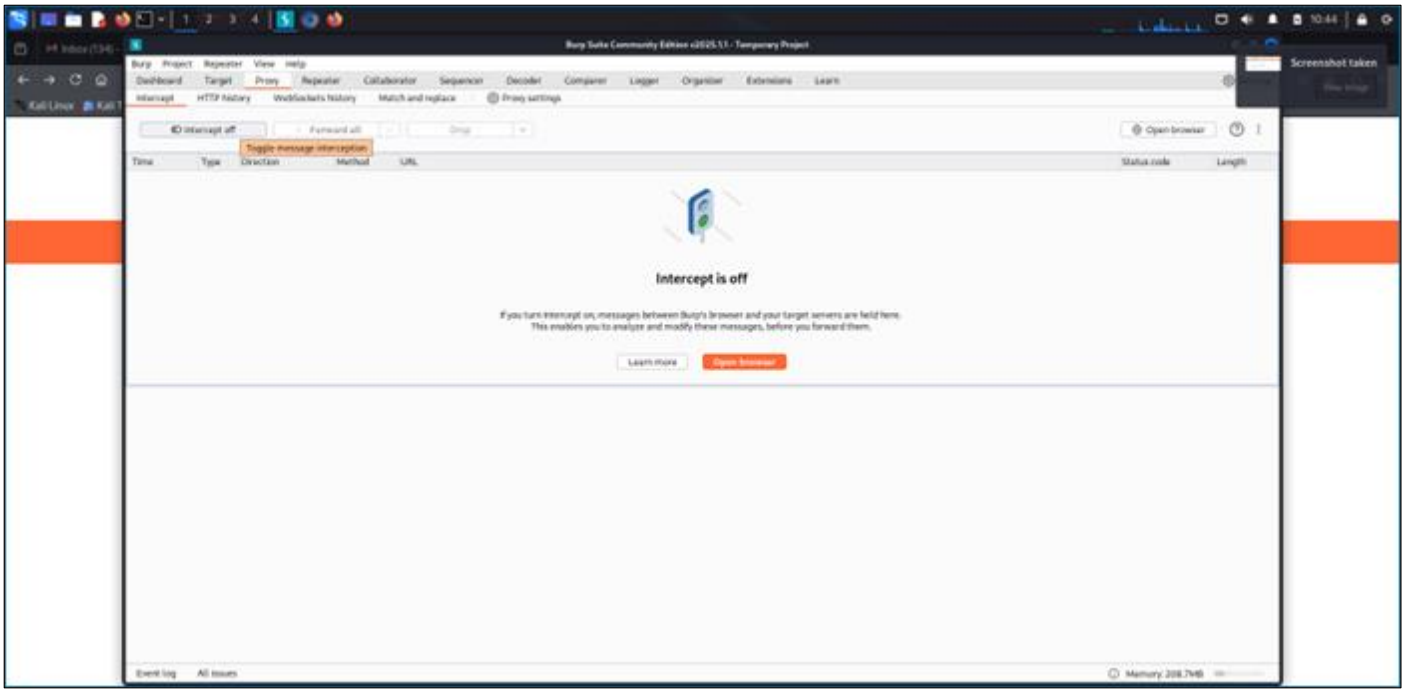
1. Open Burp Suite > Go to Proxy > Click on Open Browser.



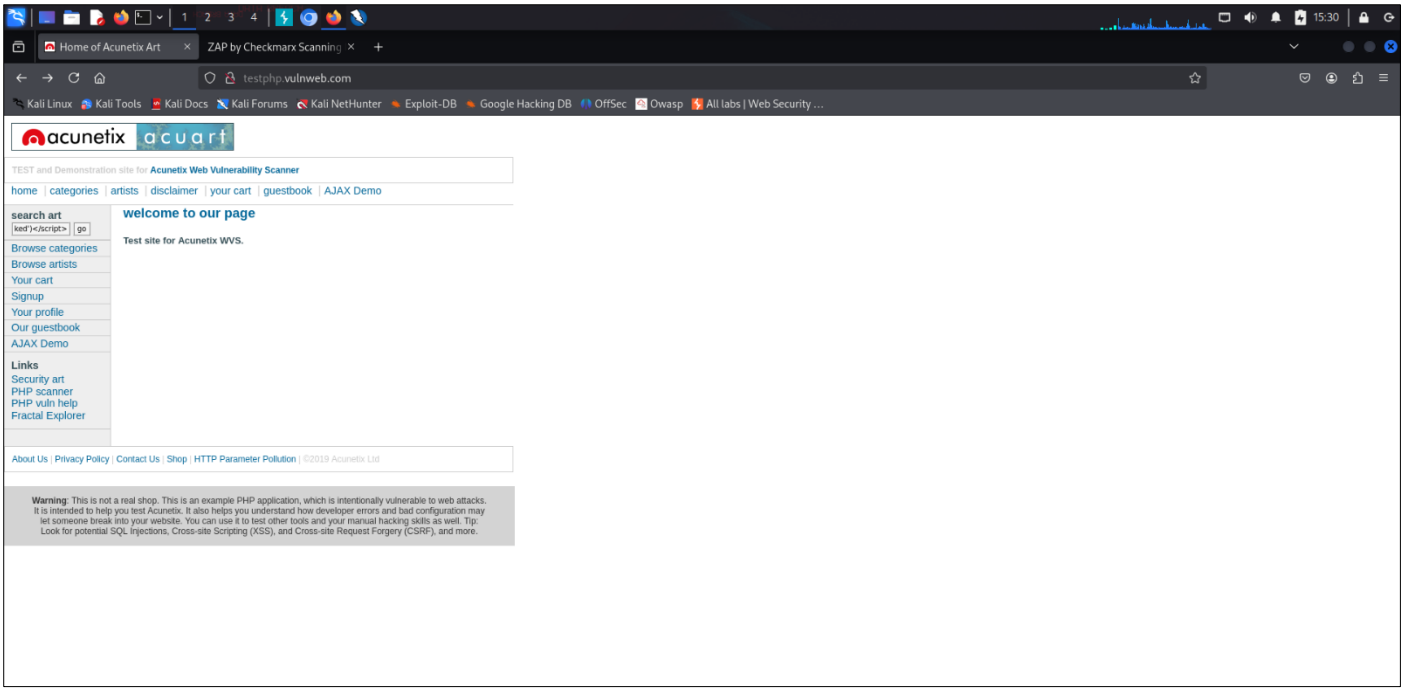
2. Access the given URL: http://testphp.vulnweb.com > write script on the search box



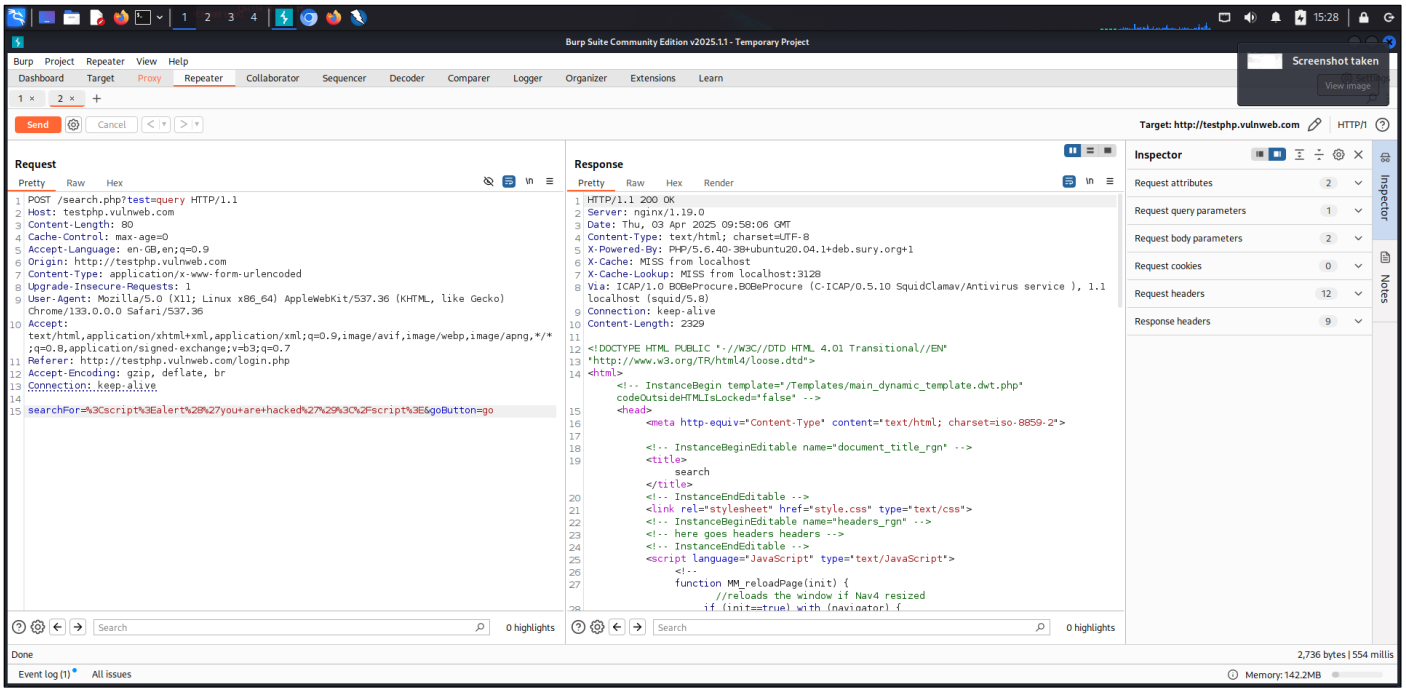
3. Before clicking Login, turn on Intercept in Burp Suite.



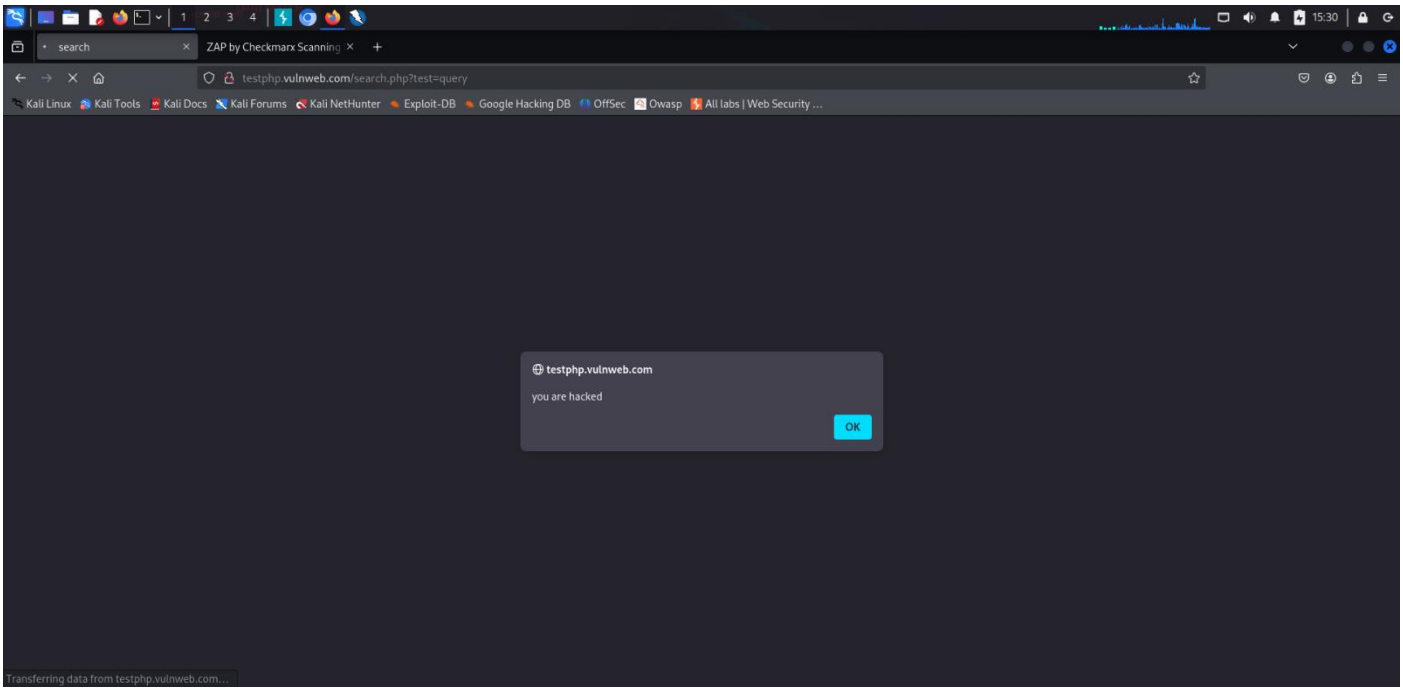
4. Click on Go



5. capture the POST request > Right-click on the request and select Send to Repeater > click on send



6. Check on the Web Page for alert



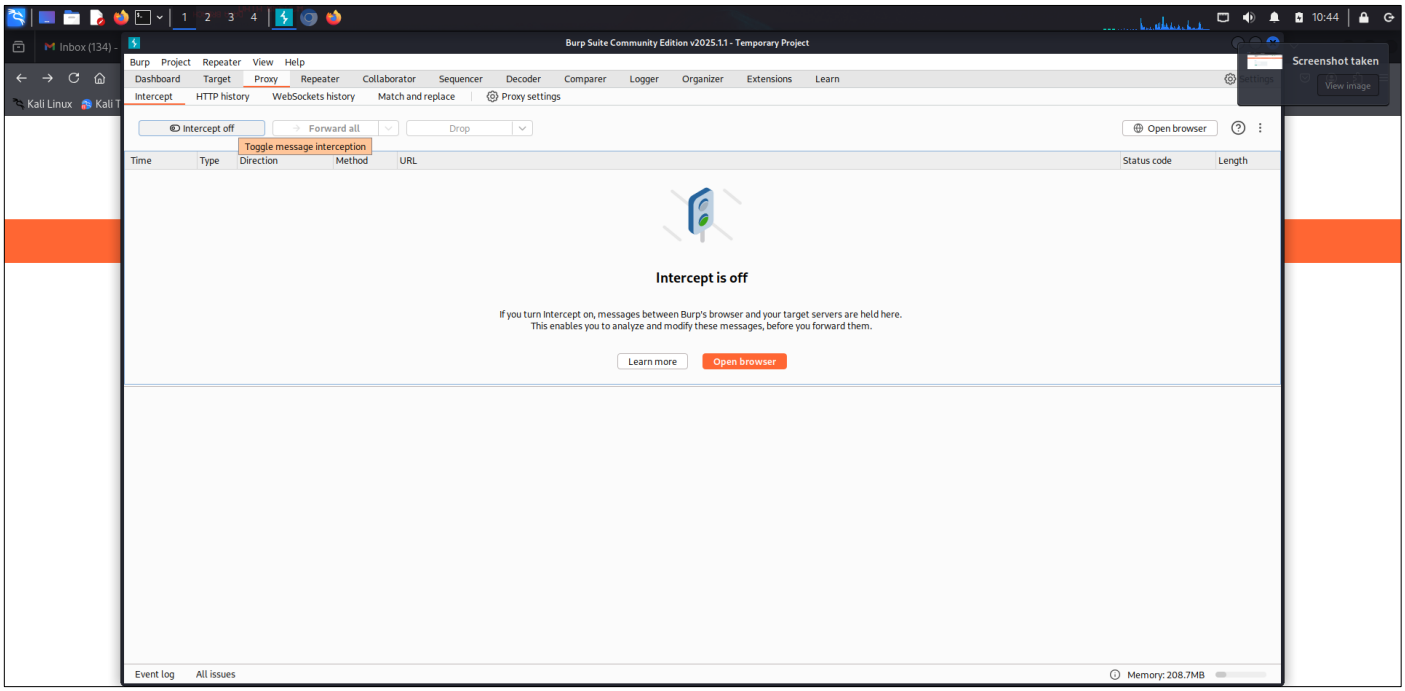
2.3 Cross-Site Request Forgery (CSRF) leads to Account Takeover

Parameter	Description
Severity	Medium
Impact	Medium
Risk Score	6.5
Affected URL	http://testphp.vulnweb.com/userinfo.php
Threat	A successful CSRF attack can result in: 1. Account Takeover – Attackers can modify user details such as email and password. 2. Data Manipulation – Attackers can alter or delete user data. 3. Privacy Violations – Sensitive user information can be exposed.

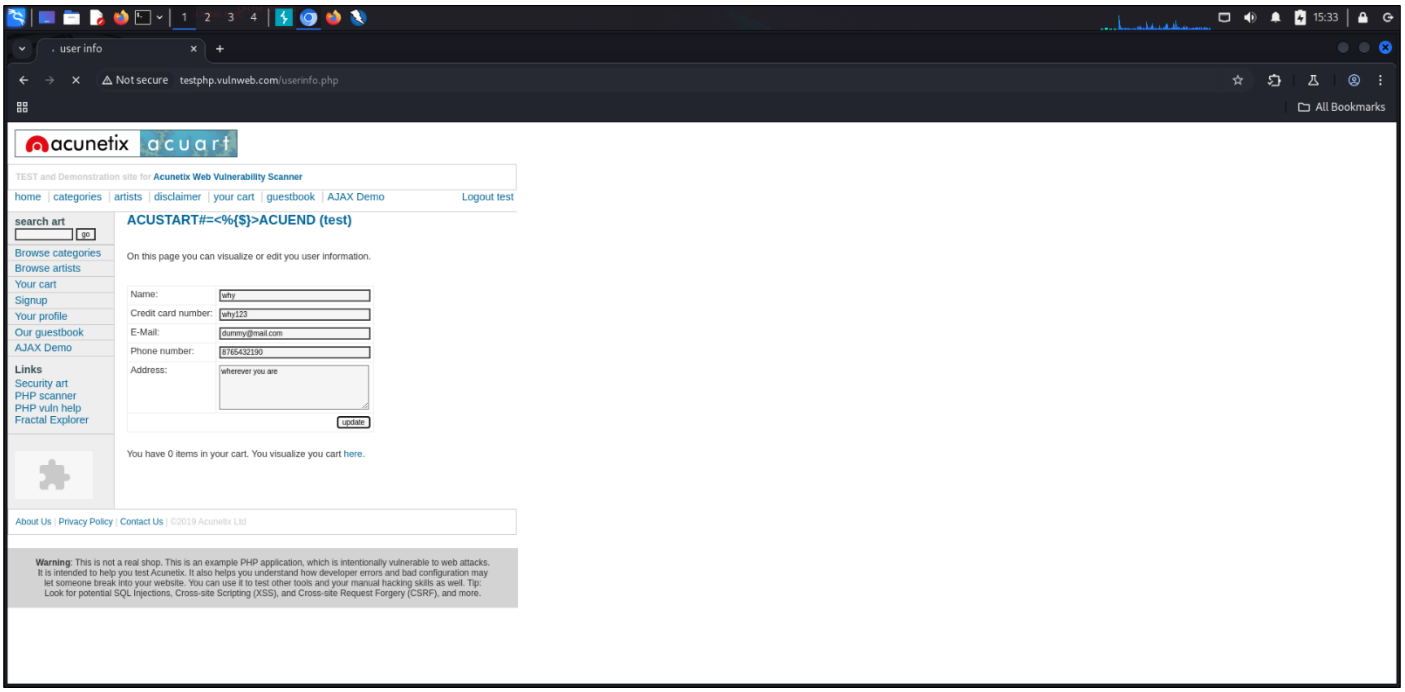
Cross-Site Request Forgery (CSRF) is a vulnerability that occurs when an attacker tricks a user's browser into making unintended requests to a website where the user is authenticated. This allows attackers to perform unauthorized actions on behalf of the user.

Steps To Reproduce

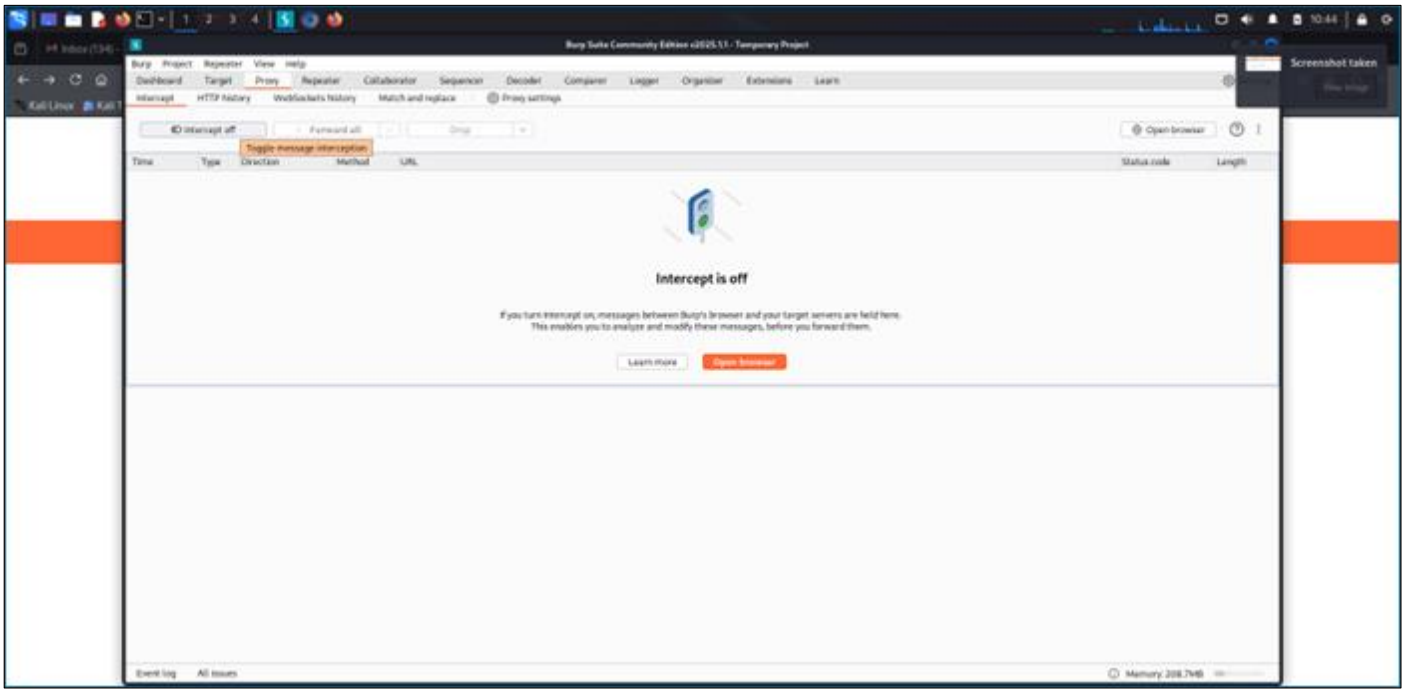
1. Open Burp Suite > Go to Proxy > Click on Open Browser.



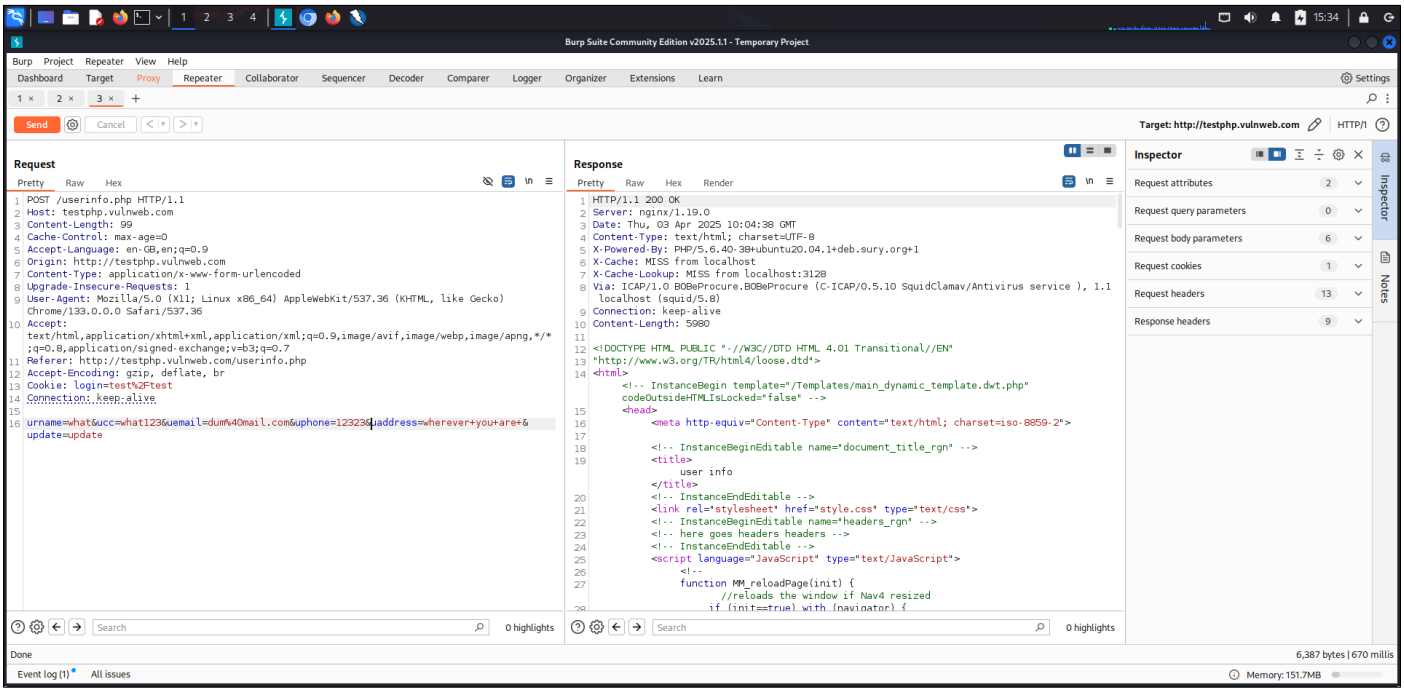
2. Access the given URL: http://testphp.vulnweb.com/userinfo.php



3. before clicking update, turn on Intercept in Burp Suite.

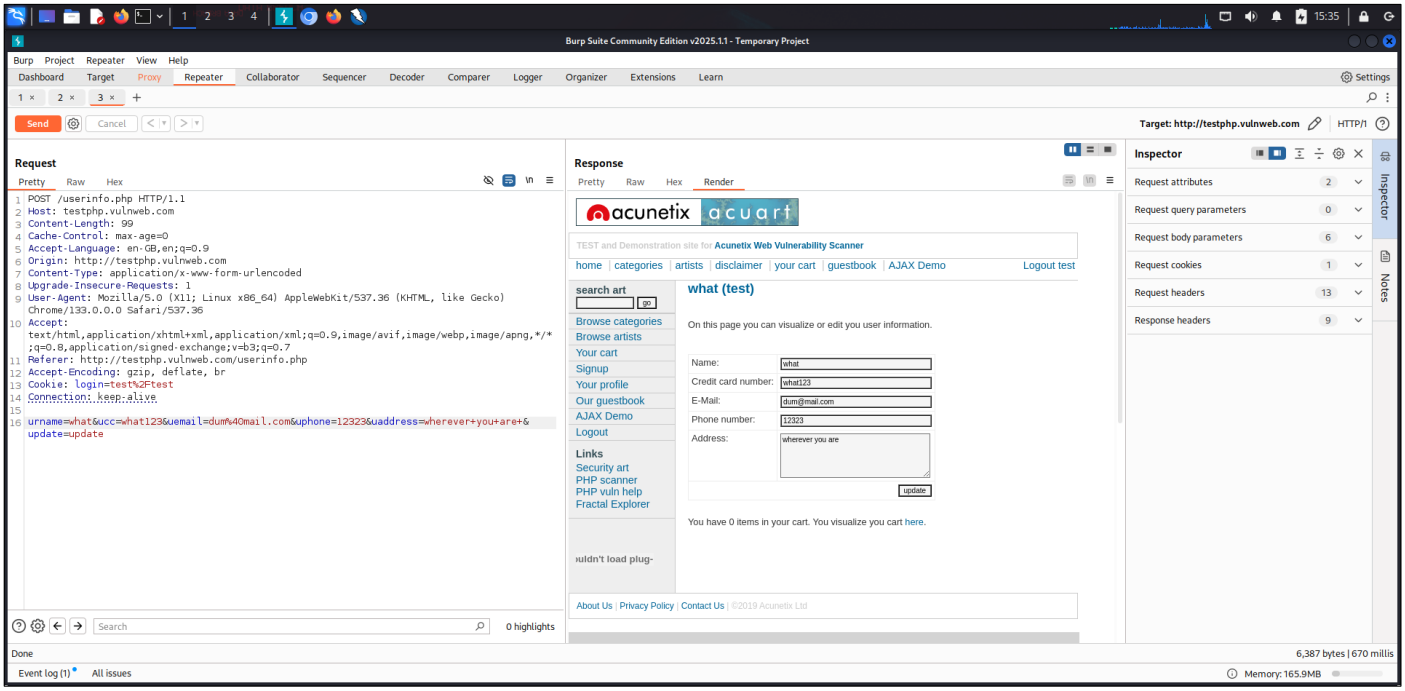


4. Click on update and capture the POST request > Right-click on the request and select Send to Repeater > In Repeater, modify the payload to user and other things you want to update



> send

5. Click Send and check the response for 200 OK > render



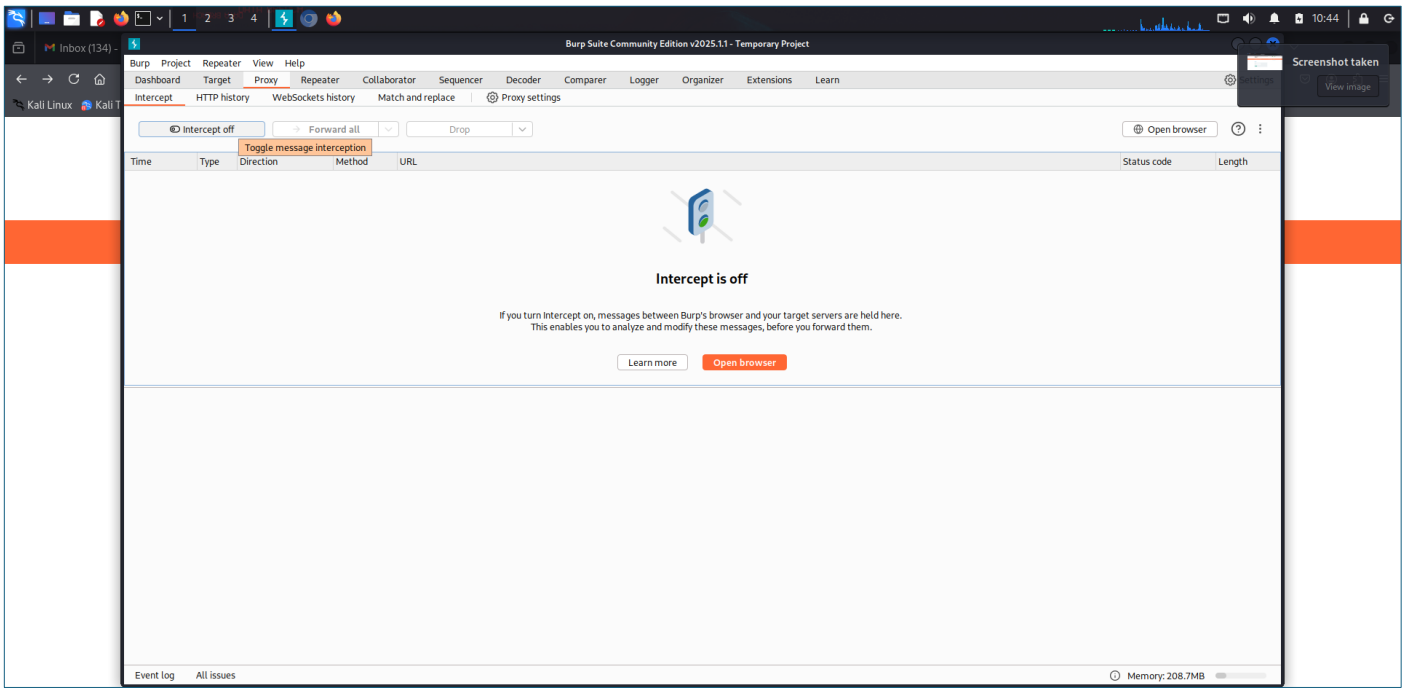
2.4 Broken Authentication leads to Unauthorized Access

Parameter	Description
Severity	Medium
Impact	Medium
Risk Score	7.3
Affected URL	http://testphp.vulnweb.com/robots.txt
Threat	A successful Broken Authentication attack can result in: 1. Account Takeover – Attackers can log in as legitimate users. 2. Data Exposure – Personal user data and sensitive information can be accessed. 3. Privilege Escalation – Attackers can gain administrative control over the system. 4. Session Hijacking – Attackers can steal and reuse active session tokens.

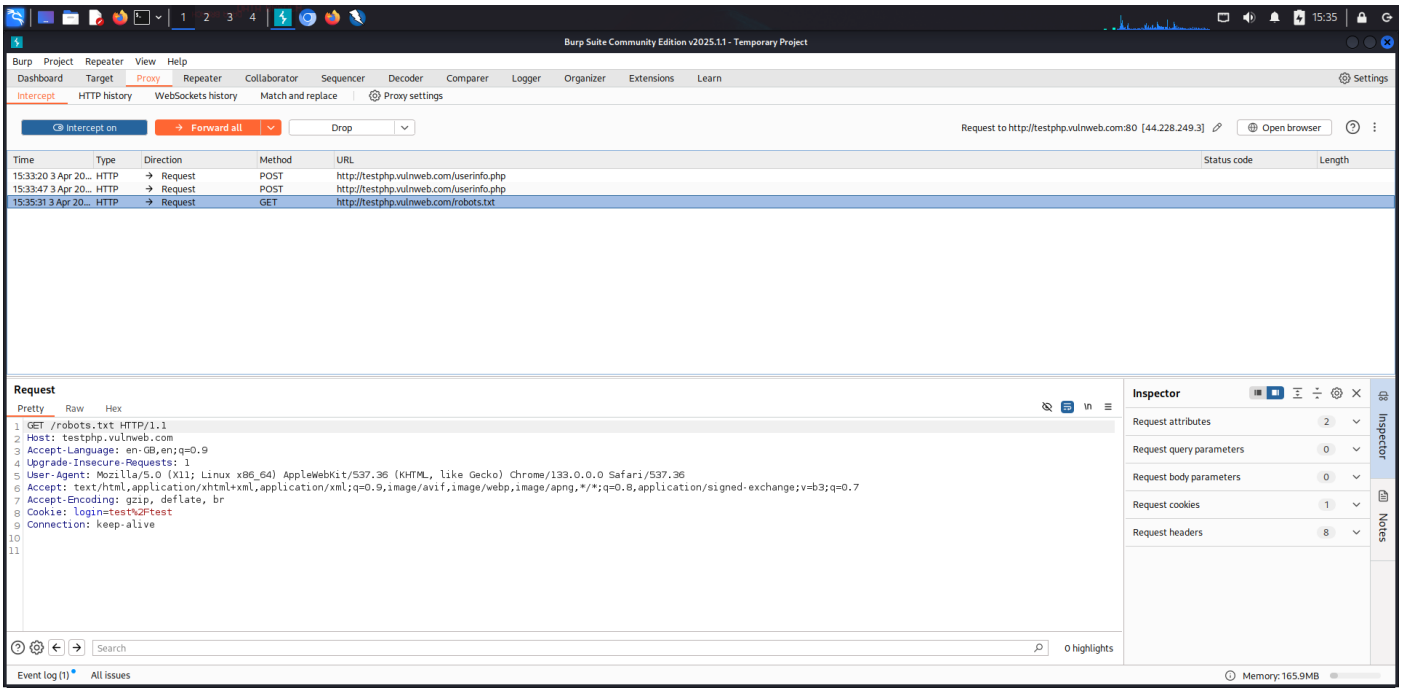
Broken Authentication occurs when an application does not properly secure user authentication mechanisms, allowing attackers to compromise accounts. This includes weak password policies, missing multi-factor authentication, and session management flaws.

Steps To Reproduce

1. Open Burp Suite > Go to Proxy > Click on Open Browser.



2. Access the given URL: http://testphp.vulnweb.com/robots.txt.



Testing Methodology

This section outlines the approach taken to assess the security posture of the target application.

Methodology Followed

1. **Reconnaissance:** Identified the target application and gathered publicly available information.
2. **Scanning:** Used automated tools like OWASP ZAP and Burp Suite to detect vulnerabilities.
3. **Manual Testing:** Performed targeted testing for SQL Injection, XSS, CSRF, and Authentication flaws.
4. **Exploitation:** Verified identified vulnerabilities by executing controlled attacks.
5. **Reporting:** Documented findings with risk ratings and remediation steps.

Common Vulnerabilities

During testing, the following common vulnerabilities were assessed:

- **SQL Injection (SQLi)** – Injection of malicious SQL queries to access unauthorized data.
- **Cross-Site Scripting (XSS)** – Injection of malicious scripts into the web application.
- **Broken Authentication** – Bypassing authentication mechanisms to gain unauthorized access.
- **Cross-Site Request Forgery (CSRF)** – Exploiting user sessions to perform unintended actions.

Risk Assessment

Each identified vulnerability has been categorized based on its potential impact:

Vulnerability	Risk Level	Impact	Likelihood
SQL Injection	Medium	Unauthorized data access	Likely
Cross-Site Scripting	Medium	Data theft, session hijacking	Possible
Broken Authentication	Medium	Account compromise	Highly Likely
CSRF	Medium	User actions without consent	Possible

Terms and Conditions

1. This report is strictly confidential and must not be shared without authorization.
2. Testing was conducted within ethical boundaries and followed responsible disclosure practices.
3. The findings in this report are based on the assessment period and may not reflect future security posture.
4. The client is responsible for implementing the recommended mitigations.