|  |  |  |  |
| --- | --- | --- | --- |
| **Date of Performance:** | **17/ 02/ 2025** | **DIV/ Batch No:** | **B2** |
| **Student Name:** | **Akshat Yadav** | **Roll No:** | **16010122221** |

| **Title:** Analysis of sample vulnerable web applications for Man-in-Middle Attack / SQL injection etc. using Burp Suite. |
| --- |

| **Objectives:** |
| --- |
| To Analyse a vulnerable web application for Man-in-Middle Attack / SQL injection using Burp Suite. |

| **Expected Outcome of Experiment:** |
| --- |
| **CO1** - Identify and analyze web attacks. |

| **New Concepts to be learned:** |
| --- |
| Burp Suit, SQLi |

| **Abstract:** |
| --- |
| Web security is a critical aspect of modern digital applications, ensuring that data transmission and storage remain secure from potential attacks. Two common types of attacks on web applications are **Man-in-the-Middle (MitM) attacks** and **SQL Injection (SQLi)**. This document discusses these attacks, their impact, and possible countermeasures |

| **Related Theory:** |
| --- |
| 1. Man-in-the-Middle Attack (MitM)Definition: A Man-in-the-Middle (MitM) attack occurs when an attacker secretly intercepts and alters the communication between two parties without their knowledge.   How It Works: ● The attacker positions themselves between the victim and the intended server.  ● The attacker can eavesdrop, modify, or inject malicious content into the communication.  ● This can be achieved through techniques like ARP spoofing, DNS spoofing, or SSL stripping.     Impact of MitM Attacks: ● Theft of sensitive information such as login credentials, banking details, or personal data.  ● Unauthorized access to web accounts.  ● Possible injection of malware into communications.   Countermeasures: ● **Encryption:** Use HTTPS and TLS to ensure secure communication.  ● **Certificate Pinning:** Ensures that a client communicates only with trusted servers.  ● **Use Secure VPNs:** Encrypts the entire network traffic to prevent eavesdropping.  ● **Enable Multi-Factor Authentication (MFA):** Adds an extra layer of security to prevent unauthorized access.  ● **Monitor Network Activity:** Identify anomalies that indicate an ongoing MitM attack.   |  |  | | --- | --- | |  |  |      2. SQL Injection (SQLi)Definition: SQL Injection is a code injection technique that allows attackers to manipulate SQL queries executed by the application’s database.   How It Works: ● Attackers inject malicious SQL code into input fields such as login forms or search bars.  ● The database executes the malicious query, allowing attackers to:  ○ Retrieve unauthorized data.  ○ Modify or delete database records.  ○ Bypass authentication mechanisms.   Impact of SQL Injection Attacks: ● Unauthorized access to sensitive data.  ● Corruption or deletion of database records.  ● Compromise of entire web applications.   Countermeasures:     ● **Use Prepared Statements:** Ensures that user input is treated as data rather than executable code.  ● **Input Validation:** Restrict user input to expected values.  ● **Least Privilege Principle:** Restrict database user permissions to minimize damage in case of an attack.  ● **Web Application Firewalls (WAF):** Detects and blocks malicious SQL queries.  ● **Regular Security Audits:** Helps identify vulnerabilities before they can be exploited.         About Burp Suite and Steps to analyse a sample vulnerable web application: (Students can write the steps what they have explored. These are sample steps.)    Analyzing vulnerable web applications using Burp Suite for attacks like Man-in-the-Middle (MitM), SQL injection, and other common vulnerabilities involves several steps. Burp Suite is a powerful web security testing tool that includes features for intercepting, analyzing, and manipulating web traffic.  Here is a step-by-step guide on how you can use Burp Suite to analyze a sample vulnerable web application: |

| **Implementation Details:** |
| --- |
| 1. Setting Up Burp Suite(Community)    * https://portswigger.net/burp/documentation/desktop/getting-started/download-a nd- install (Refer to the procedure from above documentation for the experiment. Steps below are same.)    * Configure your browser to use Burp Suite as a proxy:      + In Burp Suite, go to the Proxy tab, and make sure the Intercept feature is turned on.      * + - By default, Burp Suite listens on localhost:8080.      * + - In your browser, set up the proxy settings to send traffic through Burp Suite. In Chrome, this can be done by going to Settings → Proxy Settings or by using a tool like FoxyProxy.        1. Intercepting HTTP Traffic    * Start intercepting: Once your browser is configured to route traffic through Burp Suite, visit the vulnerable web application.      * + Burp Suite's Proxy tab will show you the HTTP requests being made from your browser. You can modify these requests before they reach the server.   + This is important for Man-in-the-Middle (MitM) Attacks where you want to modify requests or responses to inject malicious data or intercept sensitive information.          1. Finding Vulnerabilities   Burp Suite has several tools that help identify vulnerabilities in web applications.   1. SQL Injection   SQL Injection typically involves entering malicious SQL queries into input fields to manipulate the backend database.   * + Send requests to the "Repeater": In Burp Suite’s Proxy tab, right-click on the intercepted request and send it to the Repeater.   + Test inputs: Modify the parameters in the request to test for SQL injection (e.g., by entering 1' OR '1'='1 in a form field). You can also use common SQL injection payloads and see how the server responds.   + Analyze responses: If the application is vulnerable, you might get error messages like MySQL syntax error, or unexpected behavior, indicating a possible SQL injection vulnerability.     After changing         1. Cross-Site Scripting (XSS)   XSS attacks allow an attacker to inject malicious scripts into web pages viewed by others.   * + You can test for Reflected XSS by manipulating user inputs like search boxes or URL parameters.   + Send the request to Repeater and inject a script payload such as   <script>alert('XSS')</script>.   * + Look for signs in the response where the input is reflected without being sanitized, executing the script.  1. Cross-Site Request Forgery (CSRF)   CSRF attacks exploit a user's authenticated session to perform unwanted actions.   * + Burp Suite’s CSRF Tester helps in identifying possible CSRF vulnerabilities.   + Use Burp Suite’s Intruder to send automated requests to find potential CSRF vulnerabilities by manipulating parameters or session tokens.  1. Man-in-the-Middle Attacks (MitM)   A MitM attack typically involves intercepting the communication between the client and server.   * + Intercept SSL/TLS traffic: Burp Suite can intercept HTTPS traffic by installing Burp's SSL certificate in your browser.     - Go to Proxy → Options → SSL → Import Burp’s SSL certificate.   + Once set up, you can manipulate HTTPs requests/responses or look for sensitive data such as credentials or tokens being transmitted in plaintext or insecurely.  1. Using Burp Suite Features to Test for Vulnerabilities 2. Scanner (Burp Suite Professional)    * Automated vulnerability scanning: In Burp Suite Pro, you can use the Scanner tool to automatically identify common vulnerabilities like SQL injection, XSS, and CSRF.    * Simply right-click on the request, select Scan, and Burp will begin analyzing the site for vulnerabilities. 3. Intruder    * Automated Attack: Use Burp Suite’s Intruder to launch automated attacks, such as fuzzing input fields for SQL injection or testing login forms with a list of common passwords.    * Set payload positions, choose the attack type (e.g., cluster bomb, sniper, etc.), and launch it against the target. 4. Sequencer    * Session Token Analysis: Burp Suite's Sequencer tool can be used to analyze session tokens for randomness and predictability, which may indicate poor session management. 5. Reporting Vulnerabilities   After discovering vulnerabilities, it’s essential to generate reports detailing the findings:   * + Go to the Reports section in Burp Suite (Pro version).   + You can generate a comprehensive report including screenshots, attack vectors, payloads, and vulnerability descriptions.  1. Additional Tools in Burp Suite    * Decoder: Use it to decode or encode data (such as base64-encoded data) found in HTTP requests and responses.    * Comparer: Compare two pieces of data (e.g., responses before and after modifying parameters) to find differences that may indicate vulnerabilities.   Key Points:   * + Always perform testing in a safe and controlled environment (e.g., using a legal pentesting environment, CTF challenges, or a vulnerable app designed for practice like OWASP Juice Shop or DVWA).   + Don't test real websites without permission; unauthorized testing is illegal and unethical.   + Burp Suite’s free version has limited functionality, but even with it, you can still manually identify many vulnerabilities.   By following these steps and utilizing Burp Suite’s various tools, one can analyze and identify potential vulnerabilities such as Man-in-the-Middle attacks, SQL injection, and more in web applications.  XSS- |

| **Conclusion:** |
| --- |
| Analyzing vulnerabilities in web applications is crucial for ensuring security and protecting sensitive user data. Various tools, such as OWASP ZAP, Burp Suite, Nessus, Acunetix, and Netsparker, offer different approaches to identifying security flaws like Man-in-the-Middle attacks, SQL Injection, and Cross-Site Scripting (XSS).Ultimately, the choice of a vulnerability assessment tool depends on factors such as budget, ease of use, and the required depth of analysis. Regular security audits, combined with proper security measures like encryption, input validation, and the principle of least privilege, are essential to mitigate the risks posed by web security threats. |