**Task 2: Web Application Security**

Objective of the Task

The objective of this task was to understand common vulnerabilities in web applications and learn how to detect and mitigate them. Specifically, the focus was on identifying **SQL Injection (SQLi), Cross-Site Scripting (XSS), and Cross-Site Request Forgery (CSRF)** vulnerabilities using OWASP tools, and understanding best practices to secure web applications from these attacks.

Tools and Methods Used

* **WebGoat**: A deliberately insecure web application developed by OWASP for practicing and testing web application vulnerabilities.
* **OWASP ZAP (Zed Attack Proxy)**: A security tool used to intercept, analyze, and test web application traffic for vulnerabilities.
* **Web Browser with Proxy Configuration**: Used to route requests through OWASP ZAP for analysis.
* **Methodology**: Hands-on simulation of attacks, traffic interception, vulnerability exploitation, and documentation of mitigation techniques.

Steps Taken to Complete the Task

1. **Setup of Testing Environment**
   * Installed and launched **WebGoat** using Java (java -jar webgoat-server.jar).
   * Accessed WebGoat via http://localhost:8080/WebGoat.
   * Installed and configured **OWASP ZAP**, setting browser proxy to route traffic through ZAP.
2. **Exploring Vulnerabilities**
   * Navigated to the **SQL Injection module** in WebGoat.
   * Tested login input with:
   * admin' OR '1'='1

Successfully bypassed authentication, confirming vulnerability.

* + Moved to the **XSS module**.
  + Injected script payload:
  + <script>alert('XSS')</script>

This triggered a browser alert box, proving the presence of XSS.

* + Tested the **CSRF module** by creating a malicious request that executed an action without user consent. WebGoat demonstrated how CSRF exploits trust in authenticated sessions.

1. **Vulnerability Scanning with OWASP ZAP**
   * Used ZAP to capture requests while interacting with WebGoat.
   * ZAP highlighted unvalidated input fields and insecure requests, confirming the vulnerabilities.
2. **Mitigation Strategies Learned**
   * SQL Injection → Prevent using **parameterized queries (prepared statements)**.
   * XSS → Apply **input validation** and **output encoding** to sanitize user input.
   * CSRF → Use **CSRF tokens** and **SameSite cookies** to protect against unauthorized actions.

Challenges Faced and How They Were Overcome

* **Proxy Configuration Issues**: Initially, the browser traffic was not being captured by OWASP ZAP due to incorrect proxy settings. This was resolved by manually setting proxy IP as 127.0.0.1 and port 8080.
* **Understanding Attack Execution**: Some exercises in WebGoat required precise payloads. Following OWASP documentation and testing multiple payloads helped in successfully completing the tasks.
* **System Resource Usage**: Running WebGoat and ZAP simultaneously slowed down the system. This was mitigated by closing unnecessary applications and allocating more RAM to Java.

Results and Outcomes

* Successfully simulated and exploited **SQL Injection, XSS, and CSRF vulnerabilities** in WebGoat.
* Captured and analyzed web traffic using **OWASP ZAP**.
* Gained practical knowledge of how web attacks work and the risks they pose.
* Learned key mitigation techniques for securing web applications.

**3. Conclusion**

During my internship, I had the opportunity to gain hands-on experience in the field of cybersecurity, particularly focusing on **network security** and **web application security**. I learned how to perform vulnerability scanning, analyze network traffic, and simulate real-world attack scenarios using tools like **Nmap, Wireshark, WebGoat, and OWASP ZAP**.

This internship helped me strengthen my technical skills, especially in identifying vulnerabilities and applying mitigation techniques such as secure coding practices, input validation, and the use of parameterized queries. It also improved my problem-solving abilities and taught me how to approach challenges systematically.

On a professional level, I developed better documentation, reporting, and communication skills, which are essential in the cybersecurity domain. The experience boosted my confidence and provided me with a clear understanding of how cybersecurity practices are applied in real-world environments.

For future interns, I recommend staying curious, practicing consistently with hands-on tools, and asking questions whenever challenges arise. Building a habit of documenting findings and mitigation steps will also make learning more effective.

**4. Closing Remarks**

I would like to express my sincere gratitude to my mentors and team members for their continuous guidance and support throughout the internship. Their feedback and encouragement helped me progress and understand complex topics with ease.

This internship was an invaluable experience that bridged the gap between theoretical knowledge and practical application. It provided me with a solid foundation to pursue a career in cybersecurity and encouraged me to keep learning in this rapidly evolving field.

As feedback for the program, I suggest including **more collaborative exercises** where interns can work together on simulated attack-and-defense scenarios. This will help in improving teamwork skills along with technical learning.

Overall, this internship has been a rewarding journey, and I am grateful for the opportunity to be part of it.