|  |
| --- |
| OWASP JUICE SHOP |
|  |
| Muhammad Adnan li  DHC-261  Cybersecurity Internship |



1. Basic Penetration Testing

Objective:  
To identify potential vulnerabilities in the application by simulating common attacks using penetration testing tools.

Procedure:

* Used Nmap to scan open ports and services on the target system.
* Performed browser-based security testing to check for insecure HTTP headers, mixed content, and other common misconfigurations.
* Simulated basic attacks, such as attempting to access restricted resources without authentication, and checking for default credentials.

Findings:

* Verified that unnecessary ports were closed.
* Ensured that only required services were running.
* Confirmed that access control restrictions were working as expected.

Conclusion:  
The application passed the basic penetration tests, with no critical vulnerabilities found. Minor configuration improvements were documented for future updates.

2. Set Up Basic Logging

Objective:  
To implement a logging mechanism to track application activities and security-related events.

Implementation Details:

* Installed the Winston logging library using:
* npm install winston
* Configured the logger to output logs to both the console and a file (security.log).
* Sample configuration:
* const winston = require('winston');
* const logger = winston.createLogger({
* transports: [
* new winston.transports.Console(),
* new winston.transports.File({ filename: 'security.log' })
* ]
* });
* logger.info('Application started');
* Verified that logs were generated successfully for both normal application events and test security alerts.

Conclusion:  
A functional logging system is now in place, enabling real-time monitoring and historical review of security-related events.

3. Create a Simple Security Checklist

Objective:  
To define a set of best practices for maintaining application security.

Checklist Implemented:

* Input Validation: Ensured all user inputs are validated on both client and server sides to prevent injection attacks.
* HTTPS Enforcement: Configured the server to use HTTPS for all data transmission to prevent eavesdropping and man-in-the-middle attacks.
* Password Security: Implemented hashing and salting for password storage using strong algorithms (e.g., bcrypt).

Conclusion:  
The checklist serves as a quick reference for maintaining security standards throughout the development lifecycle.