Task 3: Secure Coding Review

By Mandar Kajbaje

BSc CS Student

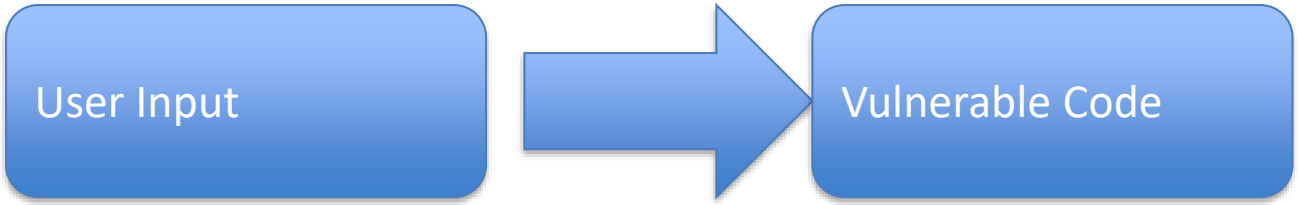
Overview

- - Selected Language: Python
- - Application Type: Simple Login System
- - Methods Used: Manual Code Review + Static Analysis
- - Goal: Identify vulnerabilities & provide remediation

Sample Vulnerable Code

- Issues Found:
- - Hardcoded credentials
- - No input validation
- - Plaintext password comparison
- - No logging or monitoring

Vulnerability Flow Diagram



Tools Used

- - Bandit (Python static analyzer)
- - PyLint security warnings
- - Manual inspection
- - OWASP Secure Coding Guidelines

Findings

- - Missing input sanitization
- - Hardcoded passwords
- - Lack of encryption
- - No error handling
- - Weak authentication logic

Recommendations

- - Use environment variables for secrets
- - Implement hashing (bcrypt/Argon2)
- - Add input validation
- - Use secure logging
- - Follow OWASP secure coding practices

Remediation Steps

- 1. Remove hardcoded credentials
- 2. Add hashing for passwords
- 3. Implement validation & error handling
- 4. Enforce secure coding rules
- 5. Re-test with static analysis tools

Thank You