Bezpieczeństwo aplikacji mobilnych

Laboratorium 1

Cel zajęć:

- Stworzenie prostej aplikacji mobilnej w Expo React Native.
- Zaprojektowanie i implementacja prostego API w wybranym języku programowania.
- Zrozumienie podstawowych zagrożeń bezpieczeństwa aplikacji mobilnych poprzez testowanie aplikacji pod kątem podatności na ataki sieciowe i aplikacyjne.

Tematyka:

- Podstawowe zagrożenia bezpieczeństwa aplikacji mobilnych.
- Ataki typu injection (np. SQL Injection, Cross-Site Scripting).
- Analiza luk w zabezpieczeniach aplikacji mobilnych.

Wymagane narzędzia:

- Node.js i npm lub yarn.
- Expo CLI.
- Emulator Android/iOS lub fizyczne urządzenie z aplikacją Expo Go.
- Wybrany język programowania do stworzenia API (np. Node.js, Python, Java).
- Narzędzia do testowania API, takie jak Postman lub Insomnia.
- OWASP ZAP lub inny skaner bezpieczeństwa aplikacji.

Zadania do wykonania:

1. **Przygotowanie środowiska:**
    - Zainstaluj Node.js oraz Expo CLI na swoim komputerze.
    - Upewnij się, że masz dostęp do emulatora Android/iOS lub fizycznego urządzenia z zainstalowaną aplikacją Expo Go.
    - Wybierz język programowania, w którym stworzysz proste API (np. Node.js, Python, Java).
    - Zainstaluj narzędzia do testowania API, takie jak Postman lub Insomnia.
2. **Stworzenie prostej aplikacji mobilnej:**
    - Utwórz nowy projekt Expo React Native z użyciem podstawowego szablonu.
    - Skonfiguruj strukturę aplikacji, dodając ekran logowania z polami dla **nazwy użytkownika** i **hasła**.
    - Dodaj przycisk **Zaloguj się**, który po kliknięciu będzie wysyłał dane logowania do Twojego API za pomocą metody **POST**.
3. **Zaprojektowanie i implementacja prostego API:**
    - Stwórz prosty serwer API w wybranym przez siebie języku programowania.
    - Zaimplementuj endpoint **/login**, który będzie przyjmował dane logowania metodą **POST**.
    - Endpoint powinien weryfikować otrzymane dane i zwracać odpowiednią odpowiedź (np. sukces lub błąd autoryzacji).
4. **Testowanie podatności na ataki typu injection:**
    - Używając aplikacji mobilnej, spróbuj wprowadzić złośliwe dane w polach logowania, np.:

        - Dla **SQL Injection**: wpisz specjalne znaki i obserwuj reakcję aplikacji oraz serwera.
        - Dla **Cross-Site Scripting (XSS)**: wprowadź ciągi znaków, które mogą wywołać niepożądane działanie po stronie klienta lub serwera.
    - Obserwuj, jak aplikacja i serwer reagują na takie dane.
5. **Analiza i identyfikacja luk w zabezpieczeniach:**
    - Sprawdź, czy Twoje API przyjmuje i przetwarza złośliwe dane bez żadnej walidacji.
    - Użyj narzędzia OWASP ZAP do przeprowadzenia skanowania aplikacji i API pod kątem podatności.
    - Zidentyfikuj potencjalne luki w zabezpieczeniach zarówno po stronie aplikacji mobilnej, jak i API.
6. **Implementacja zabezpieczeń:**
    - **Walidacja danych wejściowych:**
        - Dodaj w aplikacji mobilnej walidację pól, np. sprawdzenie, czy pola nie są puste lub nie zawierają niedozwolonych znaków.
        - Zaimplementuj walidację danych po stronie serwera, aby upewnić się, że złośliwe dane nie zostaną przetworzone.
    - **Sanityzacja danych:**
        - Upewnij się, że dane są odpowiednio oczyszczane przed ich użyciem w zapytaniach do bazy danych lub wyświetlaniem.
        - Jeśli API korzysta z bazy danych, zastosuj przygotowane zapytania (prepared statements) lub ORM w celu zapobiegania SQL Injection.