Bezpieczeństwo aplikacji mobilnych

Laboratorium 2

Analiza ruchu sieciowego i SSL/TLS

Cel:

1. Skonfigurować środowisko przy użyciu menedżera pakietów Chocolatey i narzędzia mkcert.
2. Uruchomić serwery HTTP oraz HTTPS obsługujące logowanie (POST /login).
3. Zademonstrować wyciek danych uwierzytelniających w ruchu nieszyfrowanym (HTTP) przy użyciu Wireshark i Mitmproxy.
4. Przeprowadzić inspekcję ruchu szyfrowanego (HTTPS) poprzez atak Man-in-the-Middle na emulatorze.

1) Przygotowanie środowiska i certyfikacja

1. Instalacja narzędzi (Windows)

- Upewnij się, że masz zainstalowany menedżer pakietów **Chocolatey**.
- Otwórz terminal (PowerShell/CMD) jako Administrator i zainstaluj wymagane narzędzia: choco install mkcert mitmproxy wireshark -y
- Zrestartuj terminal, aby odświeżyć zmienne środowiskowe.

2. Generowanie lokalnego Certyfikatu SSL (mkcert)

- Zainstaluj lokalny organ certyfikacji (CA) w systemie: mkcert -install
- Wygeneruj certyfikat i klucz prywatny dla localhosta (w katalogu projektu): mkcert localhost 127.0.0.1 ::1 10.0.2.2
- W katalogu powinieneś zobaczyć dwa nowe pliki (np. localhost+3.pem i localhost+3-key.pem).

3. Konfiguracja Emulatora

- Uruchom mitmproxy (lub mitmweb dla interfejsu graficznego) w terminalu.
- Skonfiguruj proxy w emulatorze Androida:
    - Settings → Network & internet → Internet → (ikona ołówka/ustawień przy WiFi/AndroidWifi).
    - Proxy: **Manual**.
    - Proxy hostname: 10.0.2.2.
    - Proxy port: 8080.

Warunek akceptacji: Narzędzia są zainstalowane, pliki certyfikatów wygenerowane w folderze projektu, a ruch z emulatora przechodzi przez Mitmproxy.

2) Konfiguracja serwerów backendowych

Cel: Uruchomienie dwóch instancji serwera (bezpiecznej i niebezpiecznej)

1. Implementacja serwera

- Stwórz prostą aplikację serwerową (np. w Node.js, Python, Go), która nie zawiera logiki biznesowej, a jedynie nasłuchuje na żądania.
- **Wymagania dla serwera:**
  - Musi posiadać endpoint: POST /login.
  - Musi przyjmować w ciele żądania (body) parametry: login oraz password.
  - Powinien zwracać odpowiedź JSON (np. { status: "logged_in" }).

2. Uruchomienie instancji HTTP

- Skonfiguruj serwer tak, aby działał na protokole HTTP na wybranym porcie (np. 3000).

3. Uruchomienie instancji HTTPS

- Skonfiguruj drugą instancję (lub ten sam serwer na innym porcie, np. 3443) tak, aby wykorzystywała wygenerowane wcześniej pliki certyfikatów (.pem i -key.pem) do szyfrowania połączenia SSL/TLS.

Warunek akceptacji:

- Aplikacja typu Postman lub cURL (uruchomiona na komputerze) potrafi połączyć się z http://localhost:3000/login oraz https://localhost:3443/login (bez błędu certyfikatu po stronie komputera, jeśli mkcert został poprawnie zainstalowany).

3) Analiza ruchu nieszyfrowanego (HTTP)

Cel: Udowodnienie, że dane przesyłane przez HTTP są widoczne dla każdego pośrednika sieciowego.

1. Wykonaj żądanie logowania (HTTP)

- W aplikacji mobilnej lub przeglądarce emulatora wykonaj żądanie POST na adres http://10.0.2.2:3000/login.
- W body prześlij testowe dane: login: student i password: tajnehaslo123.

2. Weryfikacja w Mitmproxy

- Sprawdź listę przechwyconych żądań w Mitmproxy.
- Zlokalizuj żądanie do portu 3000 i potwierdź, że widzisz treść hasła w zakładce "Request".

3. Weryfikacja w Wireshark

- Uruchom Wireshark na interfejsie sieciowym obsługującym ruch lokalny (np. Adapter Loopback).
- Ustaw filtr: `http`
- Wyślij żądanie ponownie.
- Znajdź pakiet, kliknij na niego prawym przyciskiem myszy → *Follow* → *TCP Stream* (lub *HTTP Stream*).
- Odszukaj w strumieniu przesłany login i hasło.

Warunek akceptacji: Posiadasz zrzut ekranu z Wiresharka, na którym widać hasło "tajnehaslo123" przesyłane otwartym tekstem.

4) Analiza ruchu szyfrowanego (HTTPS) i MitM

Cel: Pokazanie różnicy w analizie pakietów oraz skuteczna deszyfracja przy pomocy zaufanego certyfikatu CA (Mitmproxy).

1. Wykonaj żądanie logowania (HTTPS)

- Zmień adres w aplikacji na `https://10.0.2.2:3443/login`.
- Wyślij żądanie `POST` z danymi logowania.
- *Uwaga:* Jeśli nie zainstalowałeś jeszcze certyfikatu Mitmproxy na telefonie, połączenie może zostać odrzucone.

2. Analiza w Wireshark (Szyfrowanie)

- W Wireshark zmień filtr na port HTTPS: `tcp.port == 3443`.
- Spróbuj podejerzeć zawartość pakietów *Application Data*.
- Potwierdź, że dane są nieczytelne (zaszyfrowane).

3. Instalacja certyfikatu Mitmproxy na emulatorze

- Otwórz przeglądarkę na emulatorze i wejdź na stronę `http://mitm.it`.
- Pobierz i zainstaluj certyfikat dla systemu Android (Instalacja jako *CA Certificate* w ustawieniach bezpieczeństwa).

4. Przechwycenie i odczyt HTTPS

- Ponów żądanie `POST` na adres HTTPS.
- Sprawdź Mitmproxy: powinieneś teraz widzieć odszyfrowane nagłówki i treść body (login i hasło), mimo że połączenie jest szyfrowane TLS.
- Sprawdź Wireshark: pakiety nadal powinny być widoczne jako zaszyfrowane (TLSv1.2/1.3).

Warunek akceptacji:

1. Mitmproxy wyświetla odszyfrowane dane logowania dla połączenia HTTPS.

2. Wireshark pokazuje te same dane jako zaszyfrowany ciąg bajtów, czego dowodem jest zrzut ekranu porównawczy.