Instrukcja dla studentów — Scenariusz 4 (Lab04)

1. Pobranie projektu (GitHub)- 2pkt

Student musi sklonować repozytorium, w którym znajdują się:

- docker-compose-scenario4.yml
- Dockerfile.advanced_interceptor
- intercept_advanced.py
- foldery intercept_scripts/ i intercept_logs/

Polecenia:

git clone https://github.com/mateuszskala/BiNSC.git

cd BiNSChttps://github.com/mateuszskala/BiNSC/Lab04

2. Rola plików z repozytorium – 2pkt

Plik / folder	**Rola**
docker-compose-scenario4.yml	uruchamia 3 kontenery: klienta, proxy i serwer
Dockerfile.advanced_interceptor	buduje obraz z mitmproxy + Twoim skryptem
intercept_advanced.py	skrypt MITM – modyfikuje odpowiedzi i loguje żądania
intercept_logs/	miejsce zapisywania logów przechwyconych danych
intercept_scripts/	kopia skryptów, aby można było je modyfikować podczas zajęć

Student NIE tworzy nowych plików — korzysta tylko z tego, co jest w repo.

3. Uruchomienie środowiska -2pkt

docker compose -f docker-compose-scenario4.yml build

docker compose -f docker-compose-scenario4.yml up -d

docker ps

Powinny być 3 kontenery:

- advanced_client (172.23.0.2)
- advanced_interceptor (172.23.0.3)
- httpbin_server (172.23.0.4)

4. Generowanie ruchu z klienta -2pkt

Wejście do klienta:

docker exec -it advanced_client bash

apt update && apt install -y curl

Następnie wykonuje przesłanie url z danymi do logowania

 # GET przez proxy

curl -v -x http://172.20.0.3:8080 http://172.20.0.4:8080/get

 # POST przez proxy (z "hasłem")

curl -v -x http://172.20.0.3:8080 http://172.20.0.4:8080/post \ -d "username=ela&password=kot123"

5. Analiza logów na interceptorze -2pkt

Wykonanie skrytpu:

docker exec -it advanced_interceptor bash

cat /app/logs/requests.log

cat /app/logs/responses.log

W requests.log ma się znaleźć:

- metoda, URL, nagłówki,
- ciało żądania POST (login + hasło).

W responses.log ma być:

- status odpowiedzi,
- rozmiar odpowiedzi,
- potwierdzenie modyfikacji treści HTML.