PROGRAM SZKOLENIA APLIKACJE WEBOWE

# Szkolenie Bezpieczeństwo WWW

Przeprowadzisz najczęstsze ataki na własną aplikację i zablokujesz je.

| CZAS TRWANIA | POZIOM | FORMA |
| --- | --- | --- |
| **2 dni** | **średniozaawansowany** | **warsztat dla zespołu, zdalnie lub stacjonarnie** |

Większość wycieków danych nie wymaga wyrafinowanego ataku. Wystarczy zmienić identyfikator w adresie, wkleić skrypt w formularz albo podać ścieżkę `../` przy pobieraniu pliku. Takie błędy najłatwiej zrozumieć, gdy samodzielnie je wykorzystasz.

Na szkoleniu celowo piszesz kod podatny na konkretne ataki, atakujesz go, a potem go zabezpieczasz. Ćwiczenia wykonujesz na aplikacji w Node.js i Express, a punktem odniesienia jest lista OWASP Top 10:2025, z którą pracują audytorzy i pentesterzy. Bezpieczeństwo samego API (OWASP API Top 10) rozwijasz na szkoleniu API: REST i GraphQL, a bezpieczne publikowanie własnych paczek na szkoleniu Biblioteki npm i CLI.

## PROGRAM

**2 dni**

- Model zagrożeń i kontrola dostępu
  - OWASP Top 10:2025: co się zmieniło względem wersji z 2021 roku
  - Błędy projektowe (Insecure Design): jak je wyłapać przed kodem
  - IDOR i eskalacja uprawnień
  - Server-Side Request Forgery (SSRF)
- Uwierzytelnianie i sesje
  - Przechowywanie haseł: Argon2id lub bcrypt zamiast szybkich skrótów
  - Ciasteczka sesyjne a JWT
  - Flagi ciasteczek: `Secure`, `HttpOnly`, `SameSite`
  - Reset hasła i limity prób logowania
- Cross-Site Scripting (XSS)
  - Stored, Reflected i DOM-based
  - Kodowanie danych wyjściowych i sanityzacja HTML
  - Content Security Policy (CSP) i Trusted Types
- CSRF i clickjacking
  - Ciasteczka `SameSite` i tokeny CSRF, bo domyślne `Lax` działa tylko w Chromium
  - Nagłówki Fetch Metadata (`Sec-Fetch-Site`) jako ochrona po stronie serwera
  - Clickjacking: dyrektywa CSP `frame-ancestors` i nagłówek `X-Frame-Options`

- Wstrzykiwanie (injection) i dostęp do plików
  - SQL Injection i zapytania parametryzowane
  - Command Injection
  - Path Traversal i walidacja ścieżek do plików
- Błędy i sytuacje wyjątkowe
  - ReDoS, czyli odmowa usługi przez wyrażenie regularne
  - Obsługa wyjątków: stack trace w odpowiedzi i aplikacja, która przy błędzie przepuszcza żądanie
  - Logowanie zdarzeń bezpieczeństwa i alerty
- Transport i konfiguracja
  - Man-In-The-Middle (MITM), HTTPS i HSTS
  - Nagłówki bezpieczeństwa w Express (helmet) i błędna konfiguracja
  - CORS: jak działa i jak go nie otworzyć dla wszystkich
- Łańcuch dostaw i sekrety
  - `npm audit`, Dependabot i plik lock
  - Ataki na łańcuch dostaw: przejęte i podszywające się paczki
  - Subresource Integrity (SRI) dla skryptów z CDN
  - Sekrety w repozytorium i zmiennych środowiskowych

## DLA KOGO?

- Dla programistów frontendu i backendu aplikacji webowych.
- Dla zespołów przed audytem bezpieczeństwa lub testem penetracyjnym.

## CO MUSISZ UMIEĆ?

- Programować w JavaScript i uruchamiać aplikację w Node.js.
- Rozumieć podstawy HTTP: zapytanie, odpowiedź, nagłówki, ciasteczka.

## CZEGO SIĘ NAUCZYSZ?

- Ocenisz aplikację według OWASP Top 10:2025 i wyłapiesz błędy projektowe, zanim trafią do kodu.
- Znajdziesz i naprawisz błędy kontroli dostępu, w tym IDOR i SSRF.
- Zabezpieczysz przechowywanie haseł, logowanie, sesje i reset hasła.
- Przeprowadzisz atak XSS, CSRF i clickjacking na własną aplikację i je zablokujesz.
- Skonfigurujesz CSP, Trusted Types i nagłówki bezpieczeństwa bez psucia aplikacji.
- Napiszesz zapytania do bazy odporne na SQL Injection i zabezpieczysz ścieżki plików oraz wywołania poleceń systemowych.
- Rozpoznasz niebezpieczne wyrażenia regularne, obsłużysz błędy bez wycieku informacji i zalogujesz próby ataku.
- Ustawisz HTTPS, HSTS i CORS bez otwierania aplikacji dla wszystkich.
- Sprawdzisz zależności projektu pod kątem znanych podatności i usuniesz sekrety z repozytorium.

## OPINIE UCZESTNIKÓW

> *„Gruntownie, krok po kroku stworzyliśmy aplikację sklepu internetowego, po drodze pisząc testy jednostkowe, e2e, web komponenty, autorski backend oparty na Node.js + Express.js, jak również wykorzystaliśmy narzędzia CI."*
>
> Łukasz, JavaScript Developer (WarsawJS MasterClass)

> *„The whole team was able to learn from the mini-workshops and presentations he regularly shared with us, covering advanced Git tricks, React, JavaScript and more."*
>
> Paul Ngei, współpracownik z zespołu w Box (rekomendacja na LinkedIn)

## TRENER

### Piotr Kowalski

W branży IT od 2008 roku. Od 2014 jest mentorem dla wielu developerów. Jako trener spędził na sali szkoleniowej ponad 1000 godzin, a na LinkedIn ma 52 rekomendacje. Każde jego szkolenie to dawka solidnej wiedzy oraz humoru.

Więcej: piecioshka.pl/piotr-kowalski

## ZORGANIZUJMY TO SZKOLENIE W TWOIM ZESPOLE

Program dopasuję do Waszego projektu i poziomu grupy. Napisz, z czym pracujecie, a odpowiem z propozycją terminu i zakresu.

**KONTAKT** **piecioshka.pl/kontakt**
**E-MAIL** **piecioshka+blog [małpka] gmail.com**
**SZKOLENIE** **piecioshka.pl/szkolenia/bezpieczenstwo-aplikacji-webowej/**