CookieBanerPro

DARMOWA ŚCIĄGAWKA · LISTA KONTROLNA

Bezpieczeństwo aplikacji - 12 punktów przed uruchomieniem

Dla każdego, kto buduje aplikację lub stronę z logowaniem - także z pomocą AI. Każdy punkt ma prosty test, który zrobisz sam.

Logowanie i sekrety | Formularze i dane | Wdrożenie i repo



Przygotowane przez
CookieBanerPro
cookiebanerpro.pl

ZANIM ZACZNIESZ

Działa to jeszcze nie znaczy bezpieczne

Aplikację da się dziś zbudować w weekend, a asystent AI napisze logowanie, bazę i płatności. Rzadko jednak sam z siebie pilnuje kluczy, uprawnień i limitów. Tych luk nie widać w działaniu - wychodzą dopiero przy wycieku.

Ta ściągawka zbiera 12 rzeczy, które warto sprawdzić przed pokazaniem aplikacji światu. Nie wymaga drogich narzędzi: przy każdym punkcie jest test, który zrobisz w przeglądarce albo jednym poleceniem. Jeśli test się nie udaje - masz zadanie na dziś.

Jak korzystać

1
Przed startem
przejdź wszystkie 12 punktów, zanim wyślesz link pierwszym użytkownikom

2
Po dużej zmianie
nowe logowanie, płatności albo wgrywanie plików = powtórz punkty, których dotyczy

3
Co kwartał
paczki i historia repo (punkty 11-12) starzeją się same, nawet gdy kod stoi

Trzy grupy

1-4	**Dostęp i dane** Kto się loguje, gdzie leżą sekrety i kto widzi czyje dane
5-8	**Ataki z zewnątrz** Nie ufaj danym od użytkownika
9-12	**Uprawnienia i porządek** Serwer decyduje, a ślady nie zostają



OKRUCH RADZI

Wklej tę listę swojemu asystentowi AI i poproś: „sprawdź mój kod punkt po punkcie i pokaż, gdzie czegoś brakuje". A testy z ramek „Sprawdź" zrób już sam - one mówią prawdę.

DOSTĘP I DANE · PUNKTY 1-4

Kto się loguje, gdzie leżą sekrety i kto widzi czyje dane

1 Logowanie i hasła

- Najlepiej sprawdzona usługa logowania zamiast własnej.
- Własne logowanie: hasła tylko jako skrót z solą (argon2, bcrypt, PBKDF2). Nigdy zwykły tekst ani goły SHA.

Sprawdź: zajrzyj do tabeli użytkowników. Zamiast haseł mają być ciągi typu `$argon2id$…` albo `$2b$…`.

2 Klucze API

- Wszystkie sekrety tylko na serwerze: zmienne środowiskowe, sekrety Workera, menedżer sekretów.
- Kod strony widzi każdy. Znaleziony w nim klucz od razu wymień na nowy.

Sprawdź: otwórz stronę, Ctrl+U i narzędzia deweloperskie (Źródła). Wyszukaj `sk_`, `api_key`, `secret`, `token`.

3 Baza danych

- Każde zapytanie ograniczone do danych zalogowanego użytkownika.
- Domyślnie brak dostępu - uprawnienia dodajesz świadomie, a nie odbierasz.

Sprawdź: zaloguj się jako A i spróbuj otworzyć dane B, np. zmieniając numer w adresie albo w żądaniu. Musi się nie udać.

4 HTTPS i formularze

- Wymuszone HTTPS i przekierowanie z http.
- Każde pole sprawdzane jeszcze raz na serwerze: typ, długość, zakres i kto może je wysłać. Sprawdzanie w przeglądarce to tylko podpowiedź - łatwo je ominąć.

Sprawdź: wpisz adres z `http://` - ma przekierować. Wyślij formularz z za długą wartością z pominięciem przeglądarki (np. curl) - serwer ma odmówić.

ATAKI Z ZEWNĄTRZ · PUNKTY 5-8

Nie ufaj danym od użytkownika

5 XSS i SQL

- SQL tylko z parametrami (np. `.bind()`), nigdy sklejany z tekstu od użytkownika.
- Tekst od użytkownika nigdy nie trafia na stronę jako surowy HTML - zawsze przez escape.

Sprawdź: wpisz w pole `<b>test</b>`. Na stronie ma się pokazać dokładnie ten tekst, a nie pogrubione „test".

6 Limity prób i CSRF

- Limit prób logowania (np. 5 na minutę na konto i na IP) plus opóźnienie.
- Ochrona przed żądaniami z obcych stron: tokeny CSRF albo ciasteczka SameSite i sprawdzanie nagłówka Origin.

Sprawdź: wpisz 10 razy złe hasło z rzędu. Aplikacja ma zwolnić albo czasowo zablokować kolejne próby.

7 Ciasteczka i reset hasła

- Ciasteczko sesji: `HttpOnly`, `Secure`, `SameSite=Lax`.
- Link do resetu hasła jednorazowy, ważny ok. 15 minut.

Sprawdź: narzędzia deweloperskie, Aplikacja, Ciasteczka - sprawdź flagi. Użyj linku do resetu drugi raz - ma nie działać.

8 Wgrywanie plików

- Tylko potrzebne typy (sprawdzane po zawartości, nie po rozszerzeniu), limit rozmiaru, zmiana nazwy pliku.
- Pliki w osobnym magazynie (bucket), nigdy nie uruchamiane.

Sprawdź: zmień nazwę pliku `.php` albo `.html` na `.jpg` i spróbuj go wgrać. Ma zostać odrzucony.



CIACHO RADZI

Ciasteczko sesji to ciasteczko niezbędne - nie wymaga zgody. Analityka i piksele reklamowe już tak. Gdy aplikacja ma stronę publiczną z takimi narzędziami, baner CookieBanerPro zablokuje je do czasu zgody.

UPRAWNIENIA I PORZĄDEK · PUNKTY 9-12

Serwer decyduje, a ślady nie zostają

9 Panel admina i ceny

- Rola sprawdzana na serwerze przy każdej akcji admina - ukrycie przycisku to nie zabezpieczenie.
- Ceny i sumy liczone na serwerze z własnego cennika, nigdy z tego, co przysłała przeglądarka.

Sprawdź: jako zwykły użytkownik wywołaj adres akcji admina wprost - ma być odmowa (403). Zmień cenę w żądaniu zamówienia - serwer ma ją zignorować.

10 Tryb debug

- Na produkcji wyłączony (`NODE_ENV=production` , `DEBUG=False`).
- Użytkownik widzi krótki komunikat, szczegóły błędu trafiają tylko do prywatnych logów.

Sprawdź: wywołaj błąd: nieistniejący adres, zły parametr. Na ekranie nie może być ścieżek plików, zapytań SQL ani śladu stosu.

11 Paczki

- `npm audit` (lub `pip-audit`) i aktualizacja tego, co zgłosi.
- Usuń paczki, których nie używasz - każda to cudzy kod w Twojej aplikacji.

Sprawdź: uruchom audyt przed każdym wdrożeniem. Cel: zero zgłoszeń „high” i „critical”.

12 Historia git

- Usunięcie klucza z kodu nie usuwa go z historii repozytorium.
- Każdy klucz, który kiedykolwiek trafił do repo: unieważnij i wygeneruj nowy.

Sprawdź: `gitleaks detect` albo skanowanie sekretów na GitHubie. Wynik ma być pusty.

DO WYDRUKU

Lista kontrolna przed startem

Odhacz punkt dopiero wtedy, gdy test z ramki „Sprawdź" się powiódł.

Dostęp i dane

- [ ] **1. Logowanie i hasła**
  Najlepiej sprawdzona usługa logowania zamiast własnej.
- [ ] **2. Klucze API**
  Wszystkie sekrety tylko na serwerze: zmienne środowiskowe, sekrety Workera, menedżer sekretów.
- [ ] **3. Baza danych**
  Każde zapytanie ograniczone do danych zalogowanego użytkownika.
- [ ] **4. HTTPS i formularze**
  Wymuszone HTTPS i przekierowanie z http.

Ataki z zewnątrz

- [ ] **5. XSS i SQL**
  SQL tylko z parametrami (np. .bind()), nigdy sklejany z tekstu od użytkownika.
- [ ] **6. Limity prób i CSRF**
  Limit prób logowania (np. 5 na minutę na konto i na IP) plus opóźnienie.
- [ ] **7. Ciasteczka i reset hasła**
  Ciasteczko sesji: HttpOnly, Secure, SameSite=Lax.
- [ ] **8. Wgrywanie plików**
  Tylko potrzebne typy (sprawdzane po zawartości, nie po rozszerzeniu), limit rozmiaru, zmiana nazwy pliku.

Uprawnienia i porządek

- [ ] **9. Panel admina i ceny**
  Rola sprawdzana na serwerze przy każdej akcji admina - ukrycie przycisku to nie zabezpieczenie.
- [ ] **10. Tryb debug**
  Na produkcji wyłączony (NODE_ENV=production, DEBUG=False).
- [ ] **11. Paczki**
  npm audit (lub pip-audit) i aktualizacja tego, co zgłosi.
- [ ] **12. Historia git**
  Usunięcie klucza z kodu nie usuwa go z historii repozytorium.

Aplikacja	Data sprawdzenia	Sprawdził(a)
_____	_____	_____

CookieBanerPro

Bezpieczna aplikacja
to też szacunek dla danych

Hasła, sesje i bazy chronią dane użytkowników przed włamaniem. Druga połowa to zgoda: co strona zbiera, zanim ktoś na to pozwoli. W tym pomagamy.

Skaner zgody
co strona robi przed zgodą - za darmo
cookiebanerpro.pl/sprawdz-strone

Akademia Prywatności
darmowy kurs z certyfikatem
cookiebanerpro.pl/akademia-prywatnosci

Generator dokumentów
polityka prywatności, cookies i regulamin
cookiebanerpro.pl/generator-dokumentow

Materiały edukacyjne
e-booki, filmy i ściągawki
cookiebanerpro.pl/materialy-edukacyjne





Lista oparta na „lock" Daily Dev Drops (devdailydrops.pages.dev/lock), opracowana po polsku i uzupełniona o testy. Materiał informacyjny, nie zastępuje audytu bezpieczeństwa. Zdjęcia: Pixabay.