

Zadanie 1: Search Bar

- **Cel:** Wywołaj okienko z komunikatem (alert), używając paska wyszukiwania.
 - **Wprowadzenie:** Wiele aplikacji wyświetla wpisaną frazę (np. "Wyniki dla: ..."). Jeśli programista nie zabezpieczy tego mechanizmu, przeglądarka może pomylić Twój tekst z kodem strony.
 - **Polecenia:**
 1. Znajdź pole wyszukiwania produktów.
 2. Spróbuj wpisać tam tag HTML zawierający JavaScript
 3. Sprawdź, czy po zatwierdzeniu pojawi się okno alertu.
 - **Potwierdzenie:** Zrzut ekranu na którym widać wyskakujący alert.
-

Zadanie 2: Track Orders

- **Cel:** Wykonaj atak XSS w module śledzenia zamówień.
 - **Wprowadzenie:** Niektóre funkcje działają głównie po stronie klienta, dynamicznie wklejając dane do widoku strony.
 - **Polecenia:**
 1. Namierz funkcjonalność "Track Orders".
 2. Zamiast numeru zamówienia, wprowadź kod HTML wywołujący alert.
 3. Zatwierdź i obserwuj reakcję.
 - **Potwierdzenie:** Zrzut ekranu podstrony Track Orders na którym widać wyskakujący alert.
-

Zadanie 3: API Username

- **Cel:** Zarejestruj użytkownika ze złośliwym kodem w e-mailu, omijając blokady formularza.
- **Wprowadzenie:** Formularz na stronie może blokować znaki specjalne, ale serwer (API) często ufa danym, które do niego trafiają bezpośrednio.
- **Polecenia:**
 1. Użyj narzędzi deweloperskich.
 2. Przechwyć moment wysyłania danych rejestracyjnych.

3. Zmodyfikuj ręcznie zapytanie wysyłane do serwera – wstaw swój payload XSS w pole adresu e-mail w formacie JSON.
 4. Wyślij zmienione zapytanie.
 5. W celu weryfikacji należy zalogować się na konto administratora i wejść w panel administration
 6. Na konto administratora należy zalogować się za pomocą SQL Injection.
Formuła - w polu email: 'OR 1=1--', hasło dowolne
- **Potwierdzenie:** Zrzut ekranu z widoku panelu administration z wyskakującym alertem wynikającym z emailu użytkownika.
-

Zadanie 4: Server-side XSS Protection

- **Cel:** Omiń mechanizm czyszczący kod w formularzu opinii.
 - **Wprowadzenie:** Aplikacja usuwa niebezpieczne tagi (np. <script>). Jednak mechanizm ten ma błąd logiczny, należy go odnaleźć.
 - **Polecenia:**
 1. Przejdź do "Customer Feedback".
 2. Skonstruuj payload w polu komentarza, używając odpowiedniej techniki.
 3. Zaprojektuj go tak, aby po usunięciu środkowej części przez serwer, pozostałe fragmenty złączyły się w poprawny, złośliwy kod.
 - **Potwierdzenie:** Zrzut ekranu widoku panelu administration z alertem wywołanym przez Customer Feedback.
-

Zadanie 5: Bypass filtrów i CS

- **Cel:** Omiń filtr nazwy użytkownika oraz zabezpieczenie Content Security Policy (CSP).
- **Wprowadzenie:** Nawet jeśli wstrzykniesz kod, przeglądarka może go zablokować przez politykę bezpieczeństwa (CSP).
- **Polecenia:**
 1. **Krok 1 (Username):** Wpisz nazwę użytkownika tak, aby po usunięciu przez filtr słowa "script", z reszty liter złożył się działający tag.

2. **Krok 2 (CSP):** Wykorzystaj inne pole w profilu, aby wstrzyknąć dyrektywę osłabiającą zabezpieczenia.
- **Potwierdzenie:** Zrzut ekranu z widoku strony profilu użytkownika, na którym widać okienko alertu.

Zadanie 6: API Product Tampering

- **Cel:** Zmień nazwę produktu w sklepie, komunikując się bezpośrednio z API i pomijając interfejs graficzny.
- **Wprowadzenie:** Frontend aplikacji często ogranicza to, co użytkownik może wpisać, ale serwer (API) przyjmuje surowe dane. Jeśli endpointy odpowiedzialne za aktualizację danych (PUT/PATCH) nie są zabezpieczone, można modyfikować asortyment sklepu.
- **Polecenia:**
 1. Przeanalizuj ruch sieciowy (zakładka Network) i znajdź endpoint zwracający listę produktów.
 2. Zidentyfikuj strukturę obiektu JSON odpowiedzialnego za pojedynczy produkt.
 3. Z poziomu konsoli przeglądarki wykonaj atak, wysyłając spreparowane żądanie fetch, które nadpisze nazwę wybranego produktu (np. ID 5) na payload: `<iframe src="javascript:alert(xss)">`.
- **Potwierdzenie:** Zrzut ekranu strony głównej, na którym widać zniekształcony produkt wywołujący alert.

Zadanie 7: API Review Injection

- **Cel:** Wykorzystaj konsolę przeglądarki do wstrzyknięcia recenzji, omijając formularz na stronie.
- **Wprowadzenie:** Walidacja danych tylko po stronie klienta (w formularzu HTML/JS) nie chroni przed atakami. Atakujący może zignorować interfejs i wysłać żądanie bezpośrednio do serwera, wstawiając dane, których formularz by nie przepuścił.
- **Polecenia:**
 1. Namierz żądanie wysyłane podczas dodawania legalnej recenzji.
 2. Otwórz konsolę deweloperską i skonstruuj własne zapytanie do API.

3. Wyślij payload XSS jako treść recenzji, upewniając się, że nagłówki (Content-Type) są zgodne z wymaganiami serwera.
- **Potwierdzenie:** Zrzut ekranu podstrony produktu z widocznym alertem uruchomionym przez Twoją recenzję.
-

Zadanie 8: API BOLA & Comment Manipulation

- **Cel:** Nadpisz treść komentarza należącego do innego użytkownika.
- **Wprowadzenie:** Podatność BOLA (Broken Object Level Authorization) pozwala na manipulację obiektami, do których nie powinniśmy mieć dostępu, poprzez prostą zmianę identyfikatora (ID) w zapytaniu do API.
- **Polecenia:**
 1. Znajdź w historii zapytań sieciowych identyfikator (ID) recenzji, której autorem nie jesteś.
 2. Wykorzystując wiedzę z poprzednich zadań, przygotuj żądanie edycji (często metoda PUT lub PATCH).
 3. Zmodyfikuj zapytanie tak, aby wycelować w ID cudzej recenzji i podmienić jej treść na własny tekst lub kod.
- **Potwierdzenie:** Zrzut ekranu sekcji komentarzy, gdzie widoczna jest recenzja innego użytkownika ze zmienioną przez Ciebie treścią.