

## 16.5 XSS ін'єкції

Сайт: [ІТ курси](#)

Курс: QA manual v.5.0/2024

Книга: 16.5 XSS ін'єкції

Надруковано: Мусятовська Олена

Дата: субота 5 квітня 2025 02:48

# Зміст


[XSS ін'єкції](#)

[XSS ін'єкції](#)

[XSS ін'єкції](#)

[XSS ін'єкції: як виглядає доступ](#)

**XSS** (англ. Cross-Site Scripting — «міжсайтовий скриптинг») — досить поширена вразливість, яку можна виявити на багатьох веб-додатках. Її суть досить проста, зловмиснику вдається впровадити на сторінку JavaScript-код, який не було передбачено розробниками. Цей код буде виконуватися щоразу, коли жертви (звичайні користувачі) заходять на сторінку програми, куди цей код було додано. А далі існує кілька сценаріїв розвитку.



YOU HAVE BEEN  
HACKED !

**XSS** (англ. Cross-Site Scripting — «міжсайтовий скриптинг») — досить поширена вразливість, яку можна виявити на багатьох веб-додатках. Її суть досить проста, зловмиснику вдається впровадити на сторінку JavaScript-код, який не було передбачено розробниками. Цей код буде виконуватися щоразу, коли жертви (звичайні користувачі) заходять на сторінку програми, куди цей код було додано. А далі існує кілька сценаріїв розвитку.



YOU HAVE BEEN  
HACKED !

**Перший:** зловмиснику вдасться отримати авторизаційні дані користувача та увійти до його облікового запису.

**Друге:** зловмисник може непомітно для жертви перенаправити його на іншу сторінку-клон. Ця сторінка може виглядати цілком ідентично тій, де користувач розраховував опинитися. Але належатиме вона зловмиснику. Якщо користувач не помітить заміни і на цій сторінці введе якісь sensitive data, тобто особисті дані, вони виявляться у зловмисника.

**Третій** - майже все, що може JavaScript стає доступним для зловмисника

Однак, використання XSS-уразливостей на чужих ресурсах є незаконним.

**Перший:** зловмиснику вдасться отримати авторизаційні дані користувача та увійти до його облікового запису.

**Друге:** зловмисник може непомітно для жертви перенаправити його на іншу сторінку-клон. Ця сторінка може виглядати цілком ідентично тій, де користувач розраховував опинитися. Але належатиме вона зловмиснику. Якщо користувач не помітить заміни і на цій сторінці введе якісь sensitive data, тобто особисті дані, вони виявляться у зловмисника.

**Третій** - майже все, що може JavaScript стає доступним для зловмисника

Однак, використання XSS-уразливостей на чужих ресурсах є незаконним.

Суть у тому, що браузер не може самостійно відрізнити звичайний текст від тексту CSS, HTML або JavaScript-кодом. Він намагатиметься обробляти все, що знаходиться між тегами `<script>`, як JavaScript-код. Все, що знаходиться між тегами `<style>`, вважати CSS. І все, що схоже на тег, вважати HTML-кодом.

Якщо розробник хоче, щоб якийсь текст виглядав лише як код, але таким не був (тобто не оброблявся браузером, а виводився як є), цей текст треба спеціально обробити перш, ніж віддати його браузеру. Така обробка називається "екрануванням".

У процесі екранування тексту цьому тексті все спец. символи замінюються їх "аналогами", і браузер вже знає, напевно, що це просто текст. Найважливіше обробляти той текст, який надходить від користувача, тому що будь-який користувач може виявитися зловмисником і разом з текстом надіслати якийсь код. На жаль, іноді розробники забувають про екранування в тих чи інших місцях веб-програми, і текст виводиться без будь-якої обробки. Причин цього може бути кілька.

Суть у тому, що браузер не може самостійно відрізнити звичайний текст від тексту CSS, HTML або JavaScript-кодом. Він намагатиметься обробляти все, що знаходиться між тегами `<script>`, як JavaScript-код. Все, що знаходиться між тегами `<style>`, вважати CSS. І все, що схоже на тег, вважати HTML-кодом.

Якщо розробник хоче, щоб якийсь текст виглядав лише як код, але таким не був (тобто не оброблявся браузером, а виводився як є), цей текст треба спеціально обробити перш, ніж віддати його браузеру. Така обробка називається "екрануванням".

У процесі екранування тексту цьому тексті все спец. символи замінюються їх "аналогами", і браузер вже знає, напевно, що це просто текст. Найважливіше обробляти той текст, який надходить від користувача, тому що будь-який користувач може виявитися зловмисником і разом з текстом надіслати якийсь код. На жаль, іноді розробники забувають про екранування в тих чи інших місцях веб-програми, і текст виводиться без будь-якої обробки. Причин цього може бути кілька.

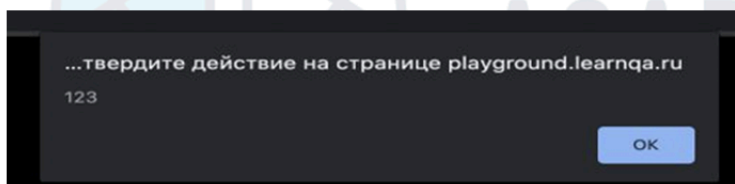
## XSS ін'єкції: як виглядає доступ

XSS ін'єкції: як виглядає доступ



Користувач заходить на сайт зі скриптом

Добрий день!  
Перевірте ваш банківський рахунок.  
`test.net?search=<script>alert(XSS)</script>`



Приклад сайту із скриптом

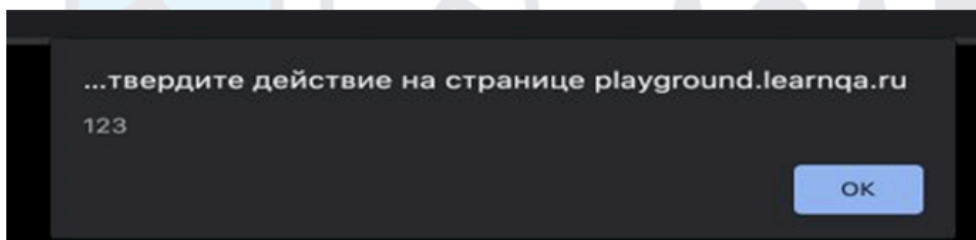
XSS демо-страница

## XSS ін'єкції: як виглядає доступ



Користувач заходить на сайт зі скриптом

Добрий день!  
Перевірте ваш банківський рахунок.  
`test.net?search=<script>alert(XSS)</script>`



Приклад сайту із скриптом

XSS демо-страница