|  |
| --- |
| Stručni kurs Razvoj bezbednog softvera |
| Izveštaj |
| Pronađene ranjivosti u projektu “RealBookStore” |

## Autor: Aleksandar Ilic

## Datum: 14.9.2025.

# Uvod

Ovaj izveštaj se bavi ranjivostima pronađenim u dole opisanoj veb aplikaciji.

## O veb aplikaciji

RealBookStore je veb aplikacija koja pruža mogućnosti pretrage, ocenjivanja i komentarisanja knjiga.

Aplikacija RealBookStore omogućava sledeće:

* Pregled i pretragu knjiga.
* Dodavanje nove knjige.
* Detaljan pregleda knjige kao i komentarisanje i ocenjivanje knjige.
* Pregled korisnika aplikacije.
* Detaljan pregled podataka korisnika.

# SQL injection

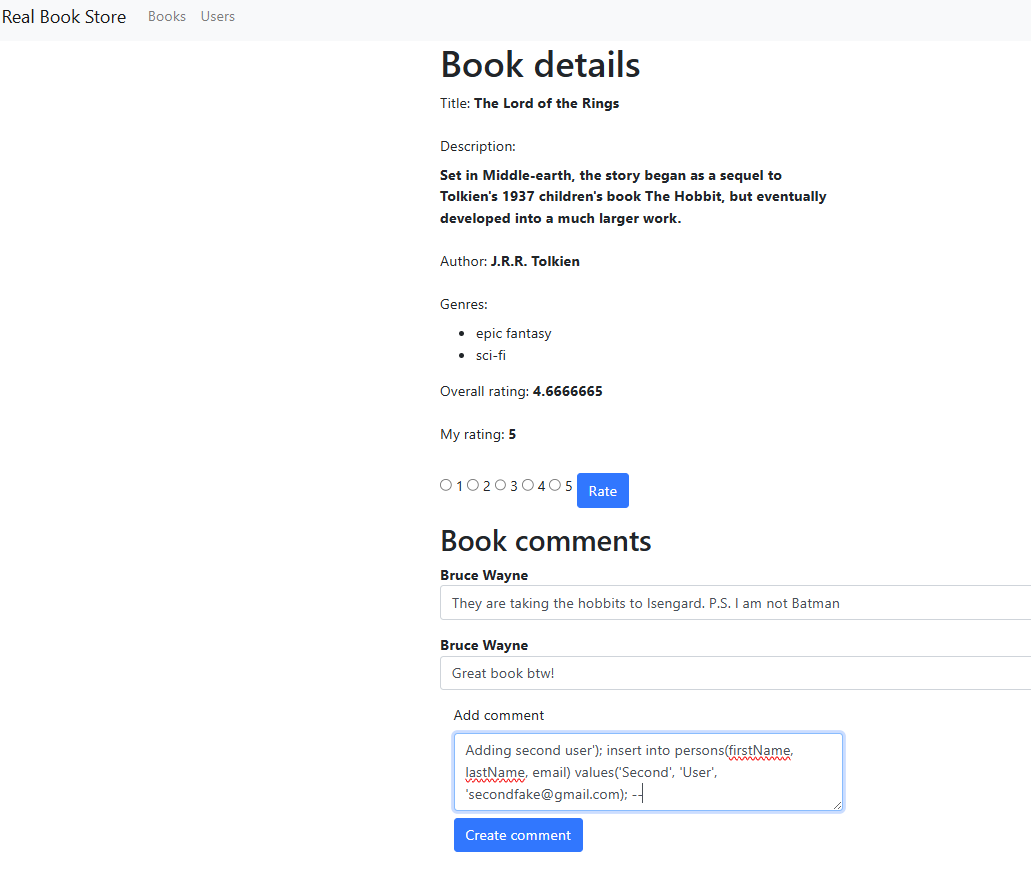
## Napad: Ubacivanje novog usera u tabelu “persons” (SQL injection)

### Metod napada:

Na stranici Books aplikacije potrebno je izabrati bilo koju knjigu i u polje “Add comment” uneti kod:

<komentar> ‘); insert into persons(firstName, lastName, email) values(‘<ime>’, ‘<prezime>’, ‘<email>’); --

Kljucna stvar je apostrof nakon komentara kojim zapravo omogucavamo da se dalji tekst nadovezuje na sql komandu u aplikaciji kao i – na kraju naseg koda jer time obezbedjujemo da ce ostatak originalne sql komande biti zakomentarisan i da nece doci do sintaksne greske. Na mesta oznacena sa <> upisati proizvoljan tekst (bez karaktera < i >).



## 

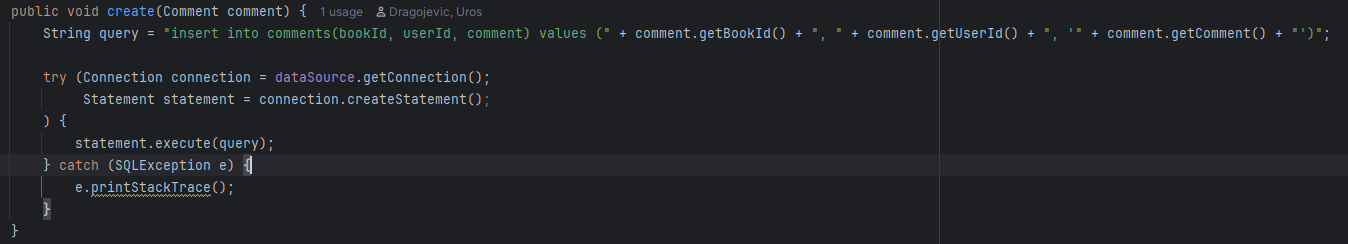
## Dodat komentar (gore) i ubacen korisnik (dole).

## 

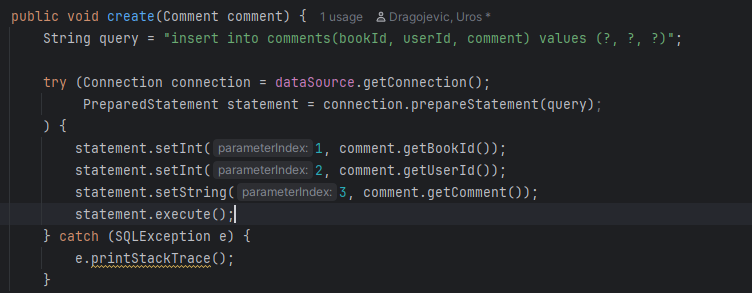
## Predlog odbrane:

Implementirati dodavanje komentara koriscenjem PreparedStatement-a.

Stari kod:



Novi kod (otporan na SQL injection):



# Cross-site scripting

## Napad: Ubacivanje novog usera u tabelu “persons”

### Metod napada:

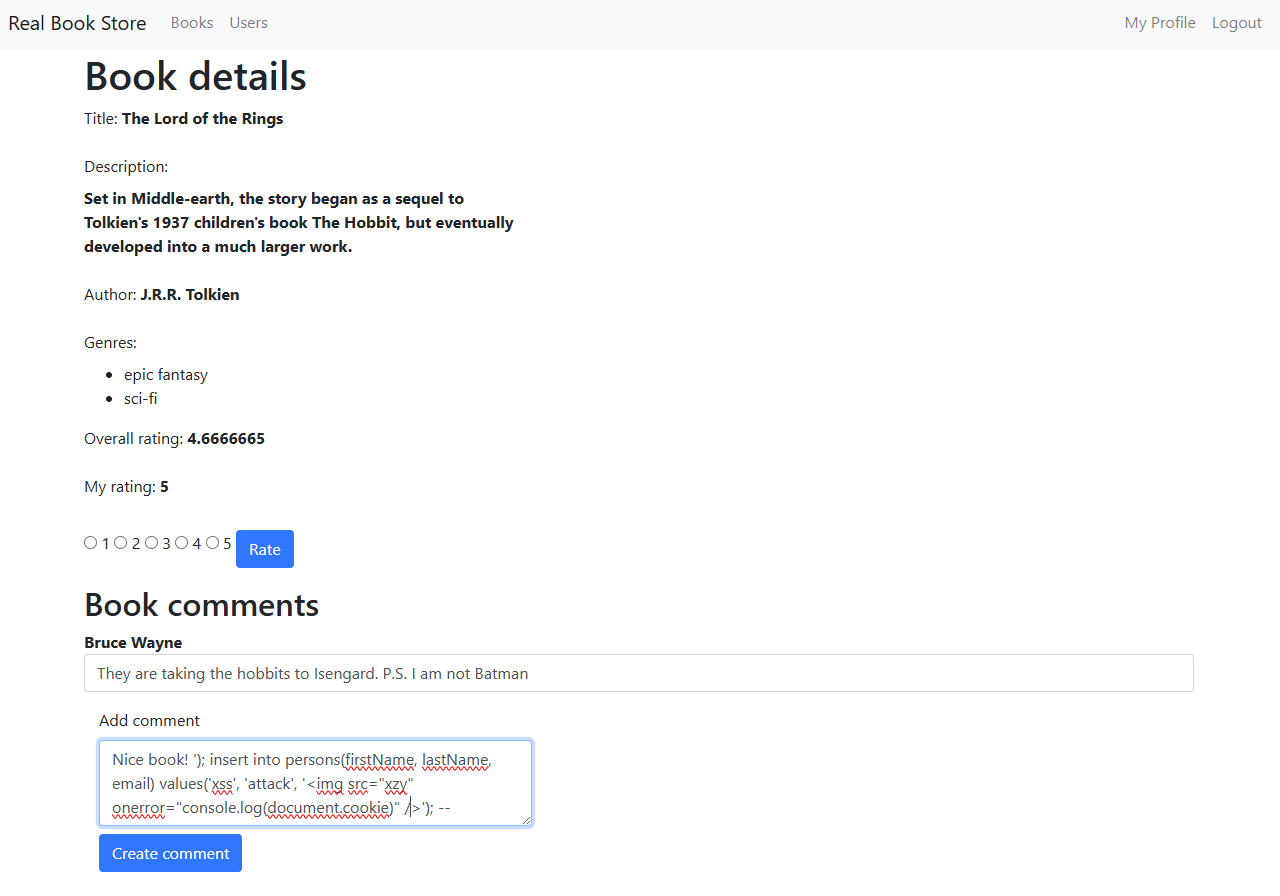
Kroz SQL injection uneti sledeci kod umesto bilo kog od atributa novo-ubacenog korisnika:

‘<img src=”xzy” onerror=”console.log(document.cookie)” />’

Odnosno ceo kod koji treba uneti u polje “Add commment” izgleda ovako:

<komentar> ‘); insert into persons(firstName, lastName, email) values(‘<ime>’, ‘<prezime>’, ‘<<img src=<nevazeca putanja do slike> onerror=”console.log(document.cookie)” />>’)

Kao sto je vec gore napomenuto payload skripta moze biti ubacena umesto bilo kog atributa user-a (ne mora biti email).



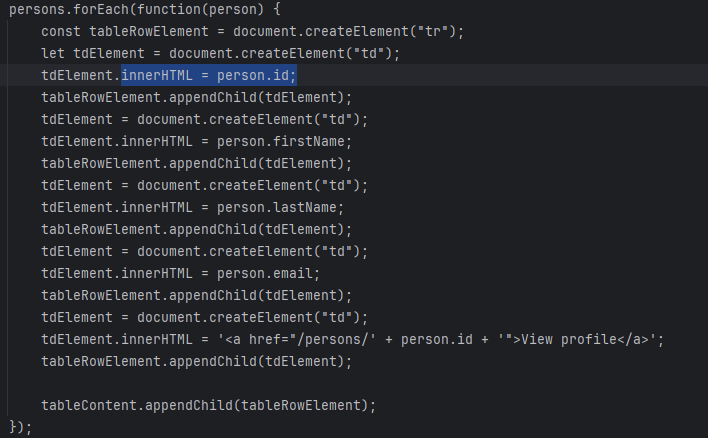
## 

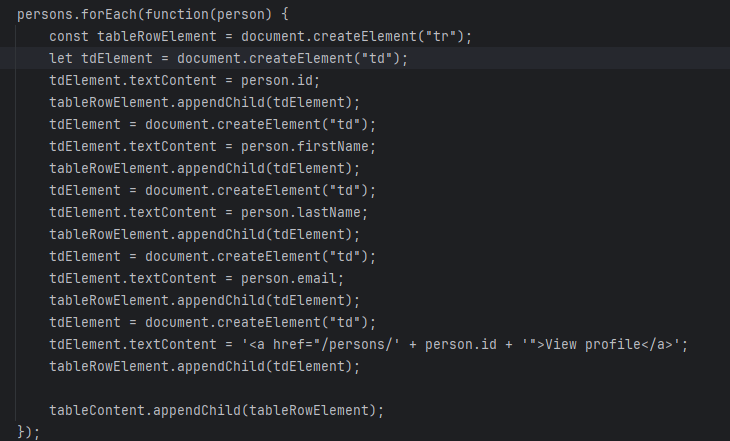
## 

## Predlog odbrane:

Implementirati search funkciju tako da koristi textContent umesto innerHTML sto omogucava da sadrzaj plain text.

Stari kod:

Novi kod (otporan na XSS kroz pretrazivanje korisnika):



Dodatno trebalo bi proveravati i oblik komentara na stranici knjige na kojoj se dodaje komentar jer je i tu moguce izvrsiti XSS napad kroz sam komentar (ovo je popravljeno zamenom th:utext u th:text u book.html u delu koji se odnosi na komentare).

# Zaključak

Treba voditi racuna o nacinu dohvatanja i cuvanja podataka koji se salju sa strane klijenta na server i obezbediti da isti mogu biti posmatrani iskljucivo kao plain text jer u suprotnom postoji opasnost zloupotrebe i podmetanja zlonamernog koda.