

18. XSS (Stored)

Disusun oleh :

Chaerul Umam, M.Kom (chaerul@dsn.dinus.ac.id)

Hafidh Akbar Sya'bani (akbar@dinustek.com)



XSS/Cross-Site Scripting (Stored)

Pada tampilan awal DVWA klik bagian XSS (Stored)

DVWA

Home
Instructions
Setup / Reset DB

Brute Force
Command Injection
CSRF
File Inclusion
File Upload
Insecure CAPTCHA
SQL Injection
SQL Injection (Blind)
Weak Session IDs
XSS (DOM)
XSS (Reflected)
XSS (Stored)
CSP Bypass

Vulnerability: Stored Cross Site Scripting (XSS)

Name *

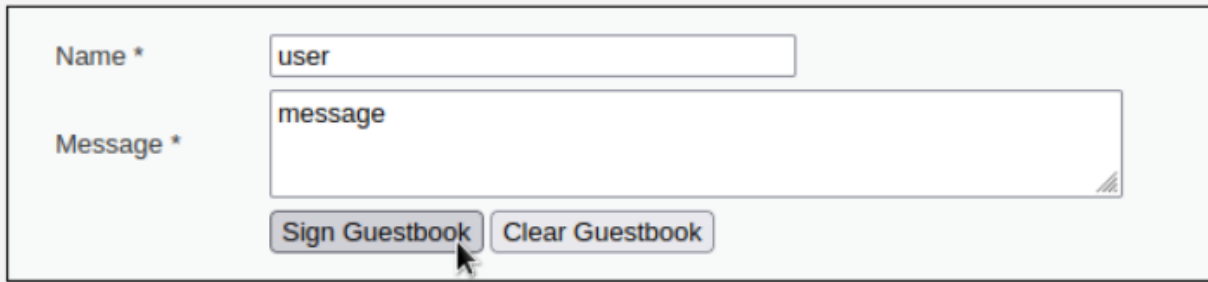
Message *

More Information

- <https://owasp.org/www-community/attacks/xss>
- <https://owasp.org/www-community/xss-filter-evasion-cheatsheet>
- https://en.wikipedia.org/wiki/Cross-site_scripting
- <http://www.cgisecurity.com/xss-faq.html>
- <http://www.scriptalert1.com/>

Coba masukkan nama dan message lalu klik Sign Guestbook

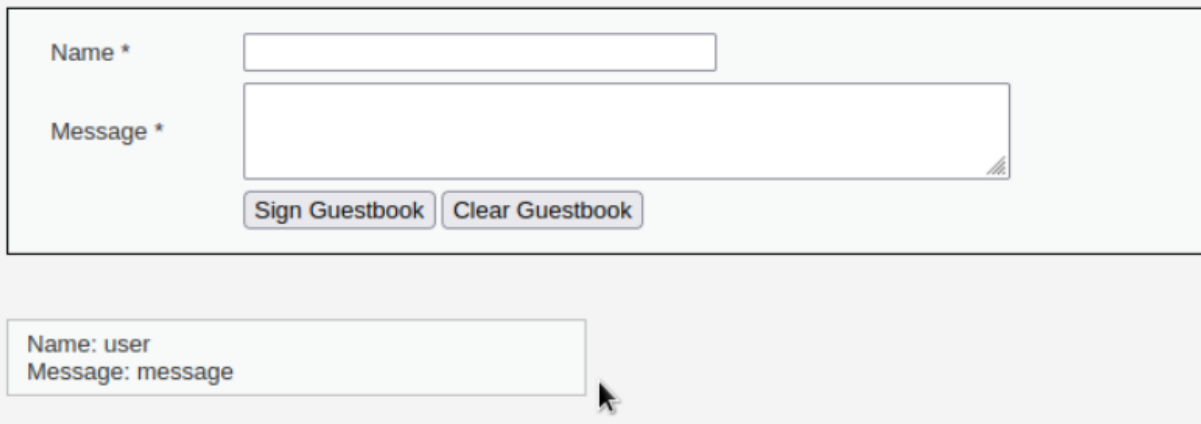
Vulnerability: Stored Cross Site Scripting (XSS)



A screenshot of a web form titled "Vulnerability: Stored Cross Site Scripting (XSS)". The form has two input fields: "Name *" with the value "user" and "Message *" with the value "message". Below the fields are two buttons: "Sign Guestbook" and "Clear Guestbook". A mouse cursor is pointing at the "Sign Guestbook" button.

Nama dan pesan tadi akan masuk dan tersimpan pada website

Vulnerability: Stored Cross Site Scripting (XSS)



A screenshot of the same web form, but now it shows the stored data. The "Name *" field contains "user" and the "Message *" field contains "message". Below the fields are two buttons: "Sign Guestbook" and "Clear Guestbook". A mouse cursor is pointing at the "Sign Guestbook" button.

Tujuan dari stored XSS adalah agar script XSS dapat tersimpan dan membuat browser dari pengguna lain menjalankan scriptnya. Script untuk stored XSS biasa dimasukkan dalam komentar, kolom chat, thread forum, bahkan form identitas seperti nama atau alamat.

Pada XSS ini coba masukkan script alert, namun karena field "Name" dibatasi jumlah karakternya, maka masukkan dalam field "Message"

```
<script>alert('xss')</script>
```

Setelah tombol Sign Guestbook ditekan, User dan Message akan dicetak pada bagian bawah, namun disini script alert tidak tereksekusi, coba untuk masukkan payload XSS lain seperti yang digunakan pada XSS sebelumnya yaitu

```
<img src/onerror=alert(1)>
```

atau

```
<input onfocus=javascript:alert(1) autofocus>
```

Namun hasilnya sama saja, kedua payload tidak dapat mengeksekusi alert

The screenshot shows a web application interface with the title "Vulnerability: Stored Cross Site". It features a form for adding a message with fields for "Name *" and "Message *", and buttons for "Sign Guestbook" and "Clear Guestbook". Below the form, there is a list of stored messages:

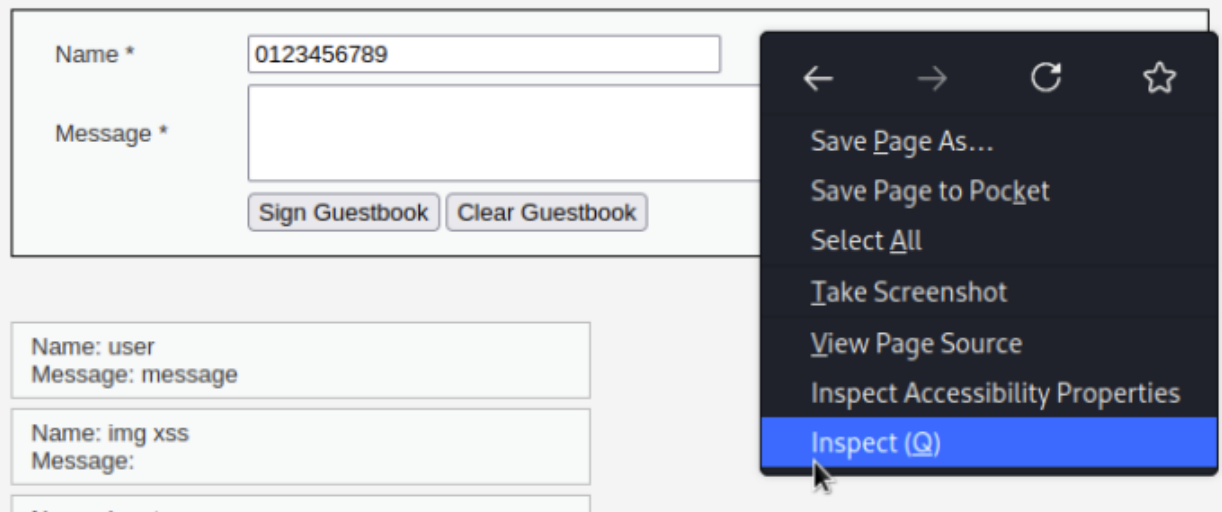
Name: user	Message: message
Name: img xss	Message:
Name: input xss	Message:

Pada kasus ini terdapat kemungkinan bahwa field Message tidak vulnerable, atau sudah diberi treatment yang baik oleh developer sehingga tidak dapat menjalankan JavaScript, jadi coba untuk gunakan field Name untuk melakukan serangan.

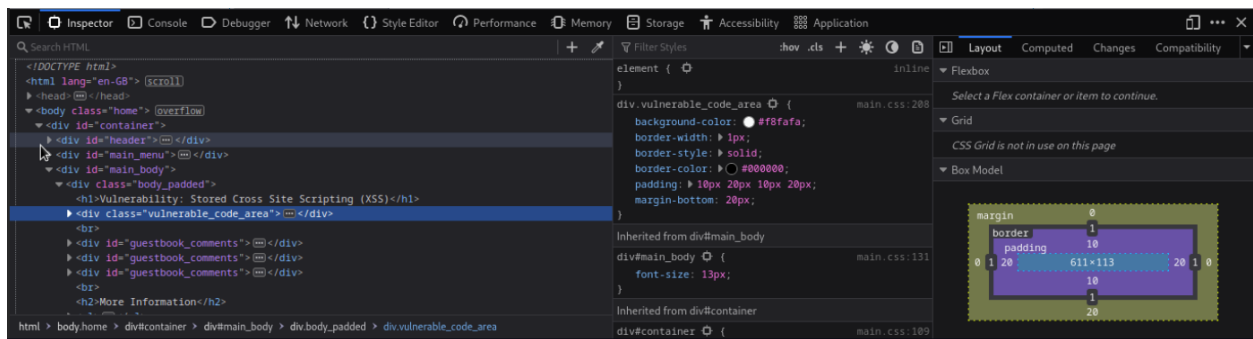
Diketahui bahwa saat ini field Name dibatasi hanya dapat menginput 10 karakter, namun batasan ini dapat diubah melalui inspect page.

Buka inspect page dengan klik kanan pada halaman DVWA lalu pilih Inspect

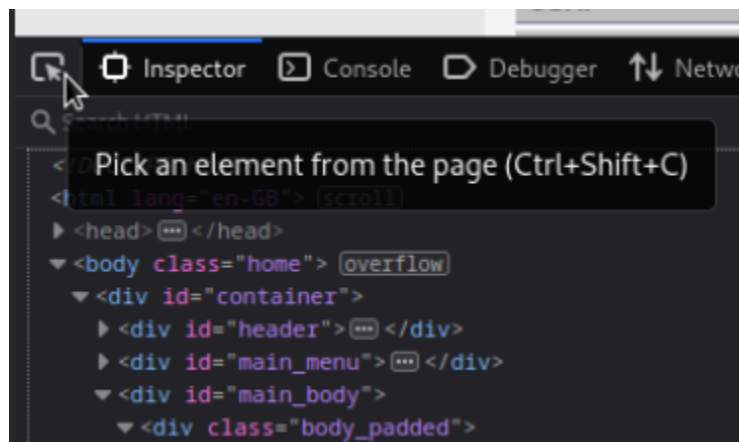
Vulnerability: Stored Cross Site Scripting (XSS)



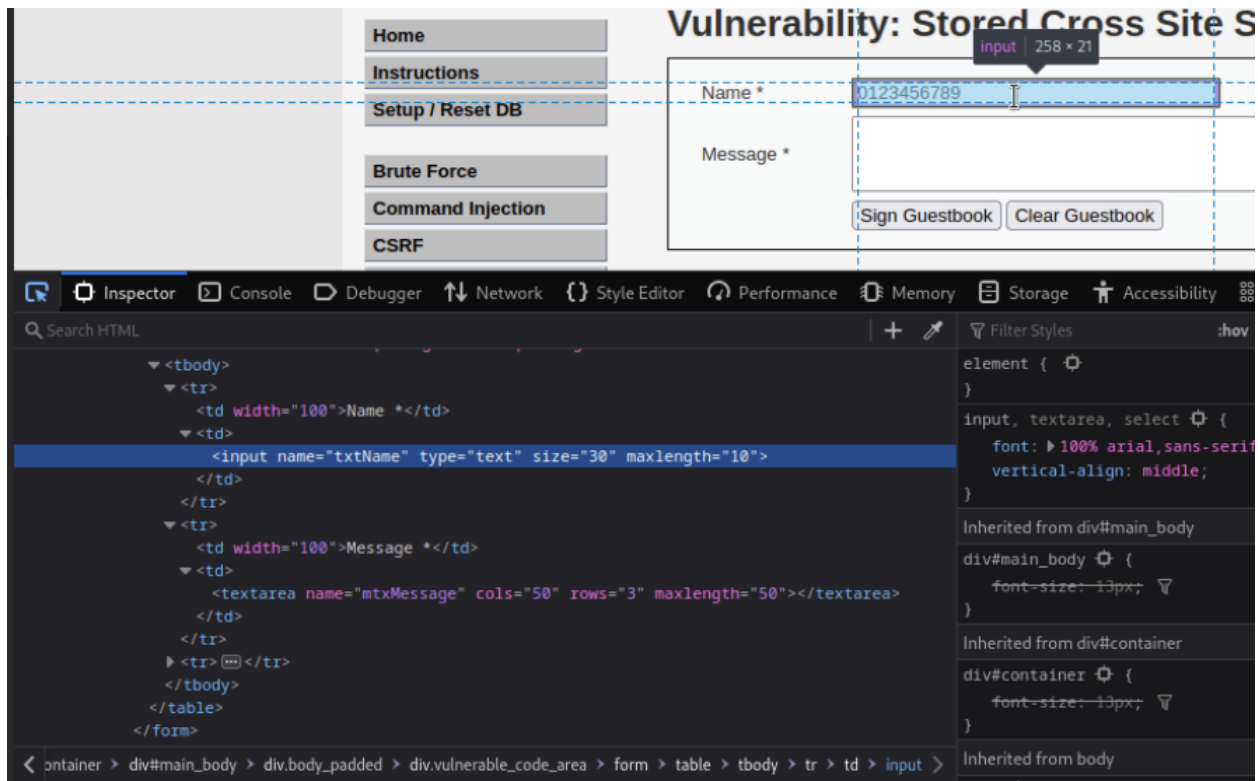
Akan terbuka panel seperti berikut



Klik pada icon cursor di sebelah pojok kiri

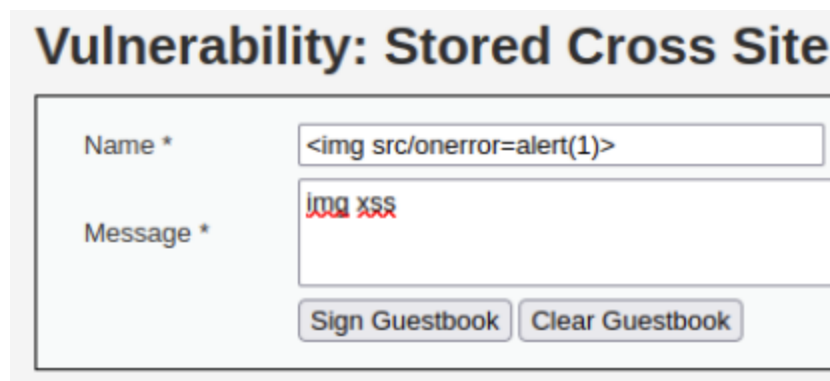


Lalu arahkan cursor mouse ke input field Name lalu klik field tersebut

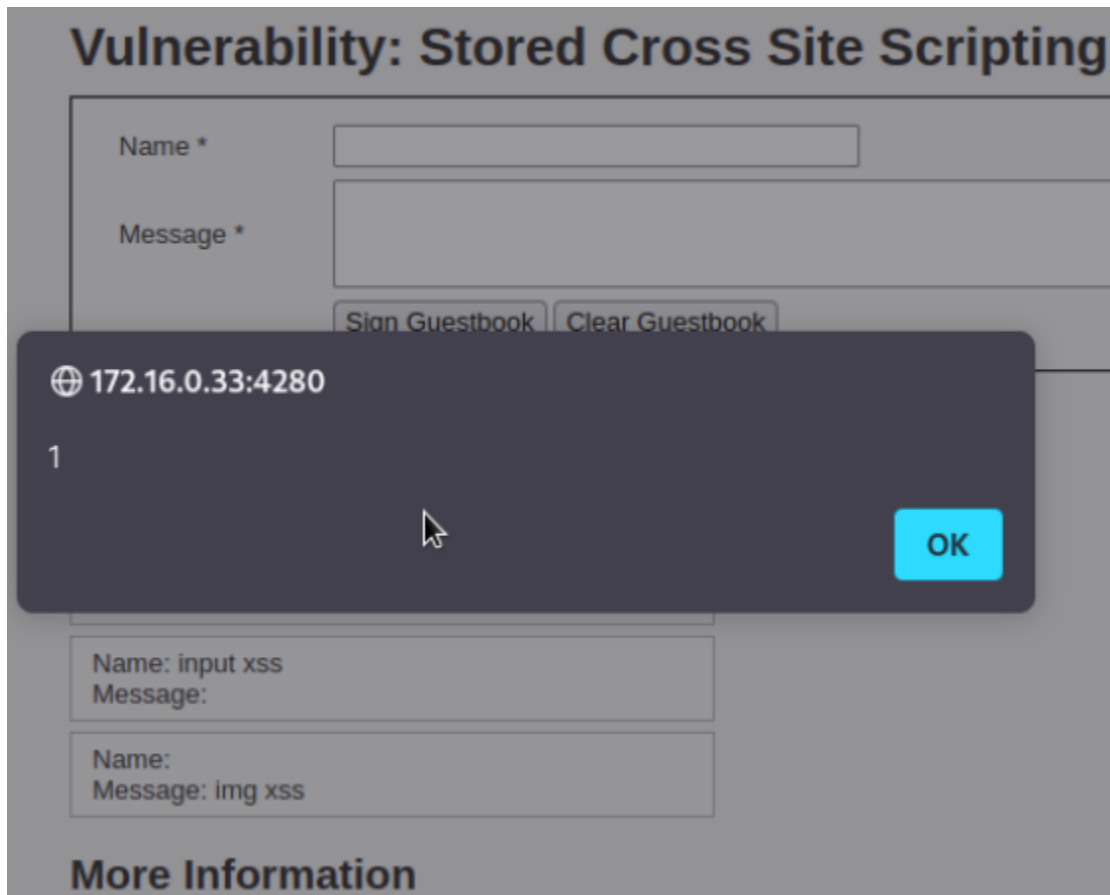


Lalu pada bagian Inspector, klik dua kali pada `maxlength="10"` dan ubah value dari 10 menjadi 100, lalu enter

Sekarang karakter maksimal pada input Name menjadi 100, tutup panel Inspect dengan klik tombol silang di pojok kanan lalu coba untuk memasukkan payload XSS pada input name



Klik Sign Guestbook



Hasilnya payload XSS berhasil dieksekusi di input field Name

Dari sini coba untuk melihat cookie browser dengan payload berikut

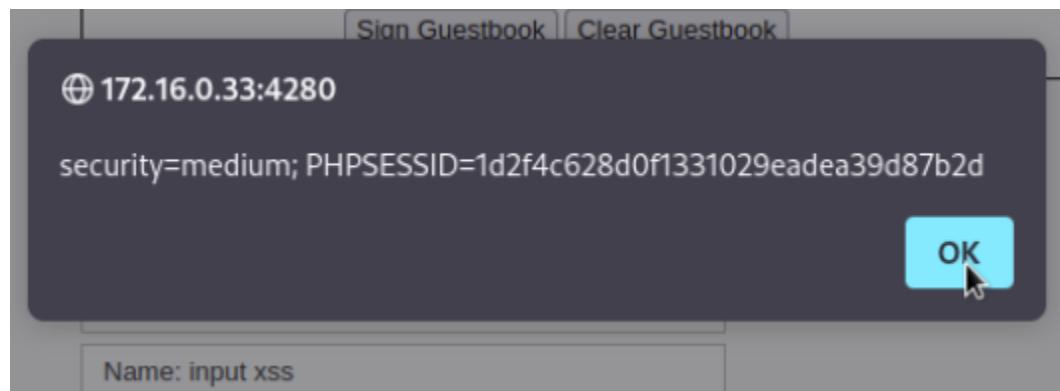
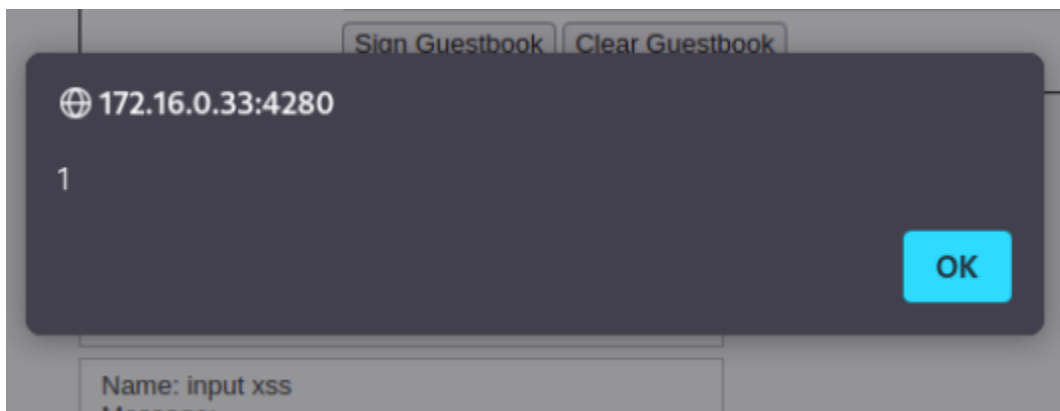
```
<img src/onerror=alert(document.cookie)>
```

Namun sebelum itu, jumlah karakter maksimal pada field Name kembali menjadi 10 karakter, ubah batasan ini menggunakan cara sebelumnya

Vulnerability: Stored Cross Site

Name *	<input type="text" value=""/>
Message *	<input type="text" value="img cookies"/>
<input type="button" value="Sign Guestbook"/> <input type="button" value="Clear Guestbook"/>	

Klik Sign Guestbook



Selanjutnya halaman akan memunculkan 2 alert, yaitu alert pertama dan alert cookie. Alert ini akan terus dieksekusi setiap page dibuka selama Message yang mengandung payload masih tersimpan.

Note

- Metode ini hanya berfungsi pada website yang vulnerable
- Beberapa website memproteksi diri dari XSS dengan melakukan block pada berbagai payload XSS sehingga hasil payload tidak akan ditampilkan
- Payload pada artikel ini hanya payload dasar dan sudah pasti banyak diblock oleh website website
- Terkadang ada beberapa payload yang belum diblock oleh website sehingga masih ada celah untuk dilakukan XSS, banyak payload yang bisa dicoba untuk melakukan XSS seperti yang ada pada [list ini](#)
- XSS DOM dan Reflected hanya akan berjalan pada browser pelaku namun tidak pada browser pengguna lain, untuk membuat XSS yang dapat berjalan di browser pengguna lain gunakan script XSS yang tersimpan (stored)