Jurnal Elektronik Ilmu Komputer Udayana
Volume 13, No 1. August 2024

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Analisis Serangan Cross Site Scripting (XSS) Pada Website OASE Menggunakan Metode OWASP

Muhammad Arrysatrya Yusuf Putranda[a1], I Komang Ari Mogi[a2], I Gusti Ngurah Anom Cahyadi Putra[a3], I Made Widiartha[a4]

[a]Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Jalan Raya Kampus Unud, Jimbaran, Bali, 80361, Indonesia
[1]arrysatrya097@student.unud.ac.id
[2]arimogi@unud.ac.id
[3]anom.cp@unud.ac.id
[4]madewidiartha@unud.ac.id

Abstract

Berkembangnya internet membuat majunya era teknologi digitalisasi, dimana hampir seluruh sektor kini dapat diakses secara digital, termasuk pendidikan. Salah satunya adalah Universitas Udayana yang memiliki LMS bernama *Online Academic Service for Elearning* (OASE) yang digunakan dalam proses pembelajaran di lingkungan Universitas Udayana. Namun perkembangan ini diikuti oleh potensi ancaman, dengan terwujudnya digitalisasi yang berarti hal tersebut dapat diakses oleh siapa saja, termasuk orang yang merusak suatu sistem. Salah satu jenis serangan yang banyak ditemukan adalah *Cross Site Scripting* (XSS). Untuk memastikan keamanan LMS OASE milik Universitas Udayana, perlu dilakukan Analisis Kerentanan terutama pada serangan XSS yang dilakukan dengan uji penetrasi menggunakan metode OWASP. Dari hasil pengujian ditemukan bahwa meskipun OASE memiliki beberapa potensi celah kerentanan, namun hanya satu fitur saja yang dikonfirmasi memiliki kerentanan, sedangkan fitur lainnya berhasil diproteksi dengan adanya fungsi *filtering* serta kontrol pada eksekusi script dari pengguna.

Kata Kunci: *Cross Site Scripting, OASE, Uji Penetrasi, OWASP, CVSS 3.1, Kerentanan*

1. Pendahuluan

Sektor-sektor tertentu mulai digitalisasi sebagai akibat dari pesatnya kemajuan teknologi, salah satunya adalah pendidikan [1]. Era digital adalah era di mana orang dapat memahami teknologi dan terhubung satu sama lain. Salah satu universitas Indonesia di Bali, Universitas Udayana, menggunakan *website* LMS *Online Academic Service for Elearning* (OASE) untuk mendukung proses pembelajaran. Sangat penting untuk menjaga keamanan data dalam era teknologi saat ini, karena banyak data yang disimpan dalam jaringan dan harus dapat diakses setiap saat.

Berdasarkan Lanskap Keamanan Siber Tahun 2022 yang dirilis oleh Badan Siber dan Sandi Negara [2]. Indonesia menempati peringkat 1 negara tujuan anomali dengan total 539.933.976. Lebih lanjut lagi berdasarkan notifikasi insiden keamanan siber yang diterima Badan Siber dan Sandi Negara, terdapat 1.433 notifikasi selama tahun 2022 dimana 336 notifikasi insiden siber diantaranya berasal dari sektor pendidikan. Terdapat 933 insiden berupa *web defacement*, 138 insiden *data breach*, serta 122 kerentanan *Cross Site Scripting*. Besarnya gap diantara top 3 indikasi insiden dari notifikasi yang dikirimkan tersebut menunjukan bahwa masih tingginya indikasi serangan siber yang mengarah pada aplikasi *website* [2].

Menurut penelitian yang dilakukan oleh Ghozali *et.al* pada 2019, peneliti melakukan penelitian untuk mendeteksi kerentanan serta menilai risiko keamanan pada aplikasi *website* menggunakan metode *Open Web Application Security Project* (OWASP), dari hasil penelitian didapatkan bahwa *website* tersebut terdapat tiga kerentanan, yakni *SQL Injection, Cross Site Scripting* (XSS), dan *Broken Authentication*. Penelitian tersebut juga menghasilkan dua faktor, yakni *impact* dan *likehood* dimana kedua faktor tersebut digunakan untuk membuat skala prioritas mengenai kerentanan mana yang harus diperbaiki [3].

Penelitian lain yang dilakukan oleh Hakim *et.al* pada 2020 mengenai serangan *Cross Site Scrpting* (XSS) berdasarkan *Base Metric* dari CVSS v.2, dimana penelitian ini mengukur tingkat kerentanan serangan XSS berdasarkan *Base Metric* CVSS v.2. Adapun hasil penelitian ini didapatkan bahwa meskipun kerentanan yang ditemukan berupa kerentanan XSS, namun dampak yang dihasilkan berbeda pada tiap *endpoint*, dimana hal ini dipengaruhi oleh beberapa faktor seperti kode pemrograman pada fitur yang diuji, *payload* pengujian yang digunakan, serta kemampuan dari penyerang [4].

Berdasarkan uraian diatas serta hasil dari beberapa penelitian terdahulu yang membahas mengenai serangan XSS, dapat dipahami bahwa kerentanan berupa XSS sangat umum pada suatu *website*. Penulis melakukan penelitian ini untuk melakukan Analisis serangan *Cross Site Scripting* (XSS) Pada *Website* OASE yang merupakan salah satu *website* penunjang proses pembelajaran pada Universitas Udayana. Adapun metode yang digunakan adalah metode *Open Web Application Security Project* (OWASP) sebagai acuan, hal ini didasari dari beberapa penelitian sebelumnya dimana metode OWASP digunakan untuk mencari kerentanan pada *website*, dan hasilnya metode ini mampu mendeteksi kerentanan yang ada, serta memberikan rekomendasi dari hasil analisis keamanan untuk meningkatkan keamanan pada sistem.

2. Metode Penelitian

Metode penelitian yang digunakan adalah metode kualitatif dimana metode ini menggunakan metode kualitatif untuk menentukan kerentanan yang ditemukan serta analisis dampak kerentanan tersebut. Lebih lanjut lagi penelitian ini menggunakan metode OWASP dalam melakukan pengujian, yang terdiri dari *Information Gathering, Penetration Testing, Report and Evaluation.*

2.1 Kajian Pustaka

a. Vulnerability

Vulnerability atau kerentanan merupakan kelemahan atau celah yang terdapat pada *website* maupun sistem yang dapat dimanfaatkan untuk merusak sistem maupun mencuri informasi. Kerentanan dapat terjadi pada *software*, sistem operasi,modul, sampai pengguna yang menggunakannya [5]. *Vulnerability* dapat dideteksi dan ditemukan melalui pemindaian kerentanan serta uji penetrasi.

b. Cross Site Scripting (XSS)

Cross Site Scripting atau XSS merupakan salah satu dari 10 jenis kerentanan yang masuk ke dalam OWASP Top 10 Vulnerability 2017. Serangan ini dilakukan dengan cara menginjeksi *script* berbahaya ke dalam suatu *website* yang memungkinkan penyerang mencuri data ataupun merusak sistem yang ada [6]. Serangan XSS selain berdampak kepada pengguna, juga berdampak kepada sistem, terdapat 3 jenis serangan XSS yakni *Stored* dan *Reflected* XSS yang memanfaatkan kerentanan pada sisi server, serta *DOM-Based* XSS yang memanfaatkan kerentanan pada sisi cilent.

c. Open Web Application Security Project (OWASP)

Open Web Application Security Project (OWASP) merupakan sebuah organisasi yang berfokus pada peningkatan keamanan *software* aplikasi *website*. Dimana tujuan utama dari organisasi ini adalah mengedukasi, mengembangkan, dan menyediakan sumber daya bagi industri serta komunitas teknologi informasi dalam upaya peningkatan keamanan *software* aplikasi *website* [7]. OWASP juga memiliki daftar top 10 kerentanan pada *website* yang paling sering ditemukan, berikut ini merupakan top 10 kerentanan pada *website* yang dirilis pada tahun 2017 seperti yang dapat dilihat pada **Tabel 1**.

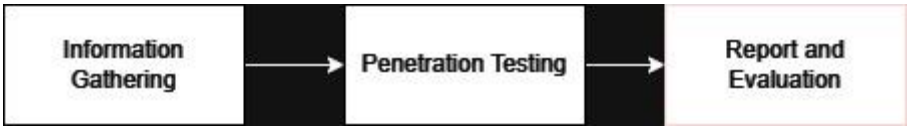
Tabel 1. OWASP Top 10 Application Security Risk 2017

A1:2017-Injection
A2:2017-Broken Authentication
A3:2017-Sensitive Data Exposure
A4:2017-XML External Entities (XXE)
A5:2017-Broken Access Control
A6:2017-Security Misconfiguration
A7:2017-Cross-Site Scripting (XSS)
A8:2017-Insecure Deserialization
A9:2017-Using Components with Known Vulnerabilities
A10:2017-Insufficient Logging & Monitoring

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2.2 Metodologi Uji Penetrasi

Penelitian ini menggunakan metode OWASP dalam melakukan uji penetrasi, dimana pengujian ini dimulai dari *Information Gathering, Penetration Testing*, dan Report and Evaluation. Adapun *flowchart* dari langkah Uji Penetrasi dapat dilihat pada **Gambar 1**.



Gambar 1. Flowchart Uji Penetrasi Metode OWASP

Seperti yang ditampilkan pada **Gambar 1**, langkah Uji Penetrasi dimulai dengan melakan *Information Gathering* berupa *endpoint* yang berpotensi memiliki celah keamanan, dalam penelitian ini seperti fitur search, chat, profile, calendar, dan course. Dimana setelah mendapatkan *endpoint* yang akan diuji, dilanjutkan dengan melakukan *Penetration Testing.*

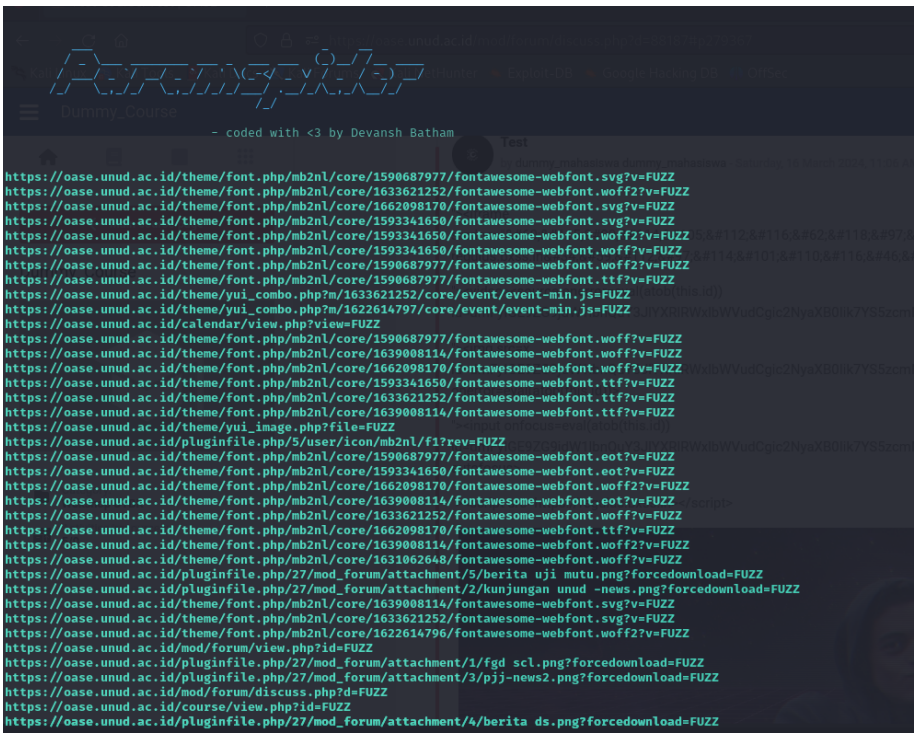
Langkah selanjutnya adalah melakukan *Penetration Testing* pada *endpoint* yang sudah didapatkan pada langkah *Information Gathering*, adapun metode ujinya adalah dengan menggunakan *payload* standar pengujian XSS pada *endpoint* tersebut, jika terdapat kerentanan XSS maka akan dilanjutkan untuk mencari sejauh apa potensi kerusakan dari kerentanan yang ditemukan.

Terakhir adalah membuat *Report and Evaluation*. Jika ditemukan kerentanan dari hasil Penetration Testing, maka dilakukan pengujian lebih lanjut untuk mencari tingkat *severity* dari kerentanan yang ditemukan tersebut, selanjutnya hasil pengujian diarsipkan dalam *Proof of Concept* yang memuat informasi mengenai kerentan yang ditemukan seperti *endpoint*, *payload, documentation, step to recreate*, serta *recommendation*. Dimana *proof of concept* ini dapat digunakan sebagai acuan bagi pemilik sistem untuk memperbaiki kerentanan yang ditemukan.

3. Hasil dan Pembahasan

3.1 Information Gathering

Pada tahapan ini dilakukan pengumpulan informasi mengenai sistem OASE, dimana ditemukan beberapa *endpoint* yang berpotensi memiliki celah kerentanan terhadap serangan XSS, diantaranya adalah fitur Chat, Search Course, Profile, Forum, Assignment, Calendar, dan Quiz. Pengumpulan informasi juga dibantu menggunakan framework *Paramspider* untuk mencari parameter yang berpotensi memiliki celah kerentanan seperti yang dapat dilihat pada **Gambar 2**.

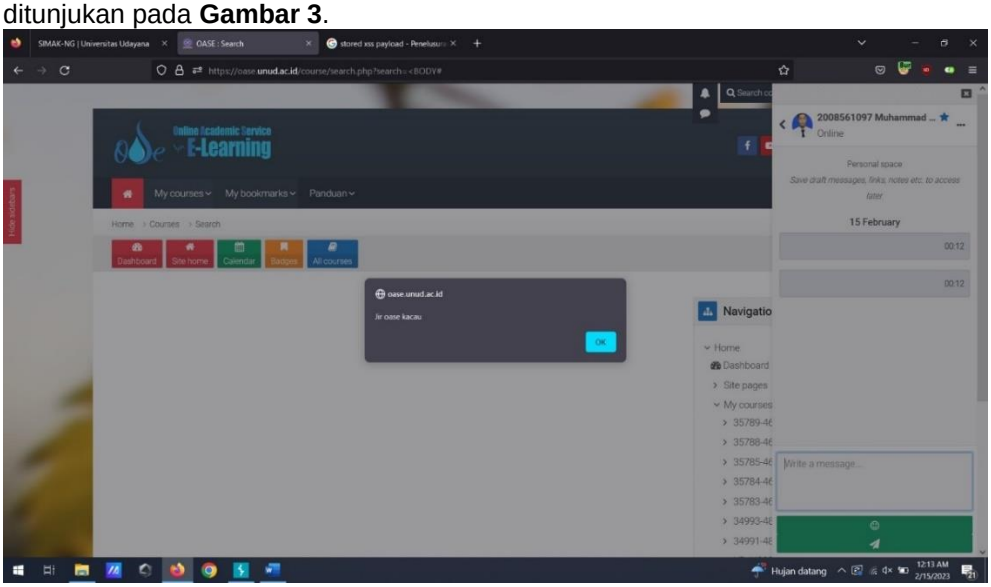


Gambar 2. Proses Pencarian Endpoint dengan Paramspider

3.2 Penetration Testing

a. Fitur Chat

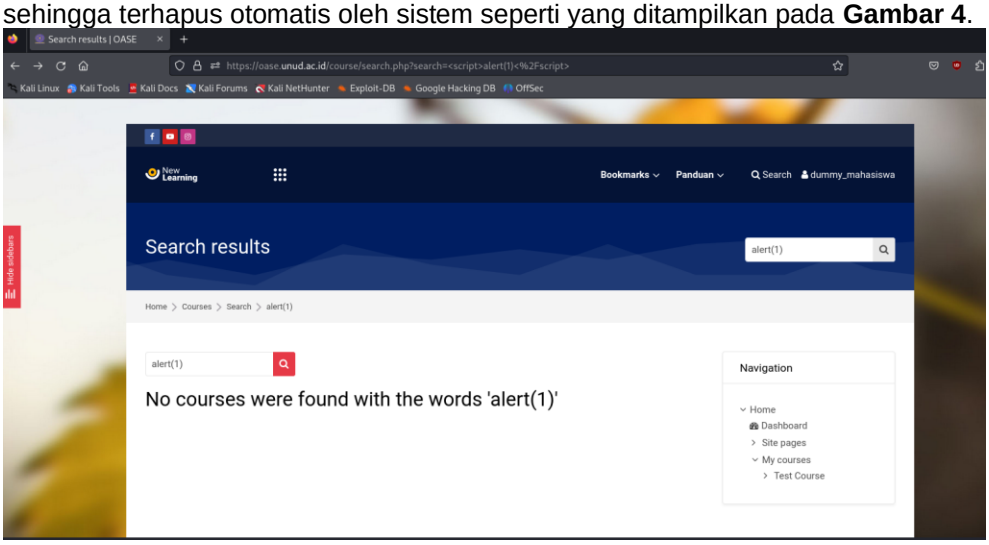
Terdapat kerentanan yang ditemukan pada fitur chat, dimana kerentanan ini memungkinan pengguna untuk mengirimkan *script* berisi *payload* XSS kepada pengguna lain kemudian *payload* tersebut akan dieksekusi oleh sistem, seperti yang ditunjukan pada **Gambar 3**.



Gambar 3. XSS Fitur Chat

b. Fitur Search

Terdapat celah kerentanan pada fitur search, dimana fungsi ini dapat menerima masukan user dari field maupun URL. Setelah dilakukan pengujian dengan *payload* standar pengujian XSS, ditemukan bahwa meskipun URL menampilkan *script* secara keseluruhan, namun fungsi *search* melakukan *filtering* kata <script> dan tanda < sehingga terhapus otomatis oleh sistem seperti yang ditampilkan pada **Gambar 4**.



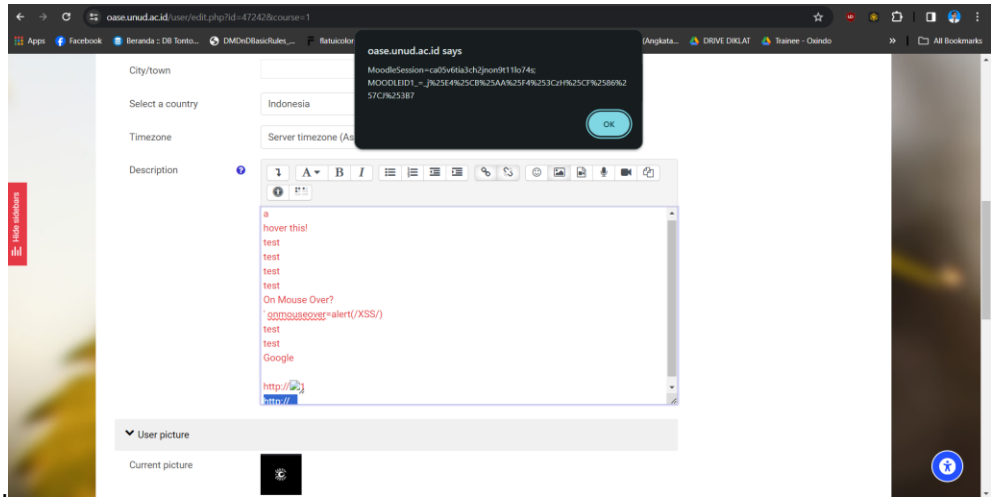
Gambar 4. Function Filtering Pada Fitur Search

Pengujian kemudian dilakukan dengan *payload* yang tidak menggunakan karakter yang difilter, hasilnya tidak ditemukan celah kerentanan karena sistem tidak mengeksekusi *script* yang terdapat pada *payload* tersebut.

c. Fitur Profile

Seperti yang ditunjukan pada **Gambar 5**, fitur profile memiliki *Field Text* yang dapat digunakan untuk memasukan *payload* serangan XSS. Pengujian dilakukan dengan menggunakan *payload* secara langsung dan dengan *hyperlink*. *Payload* yang digunakan adalah *payload* yang memasukan gambar, dimana jika terdapat *error* maka *browser* akan menampilkan *document cookie*s dari pengguna Dimana pada fitur ini dapat dilihat bahwa terdapat kerentanan serangan XSS.

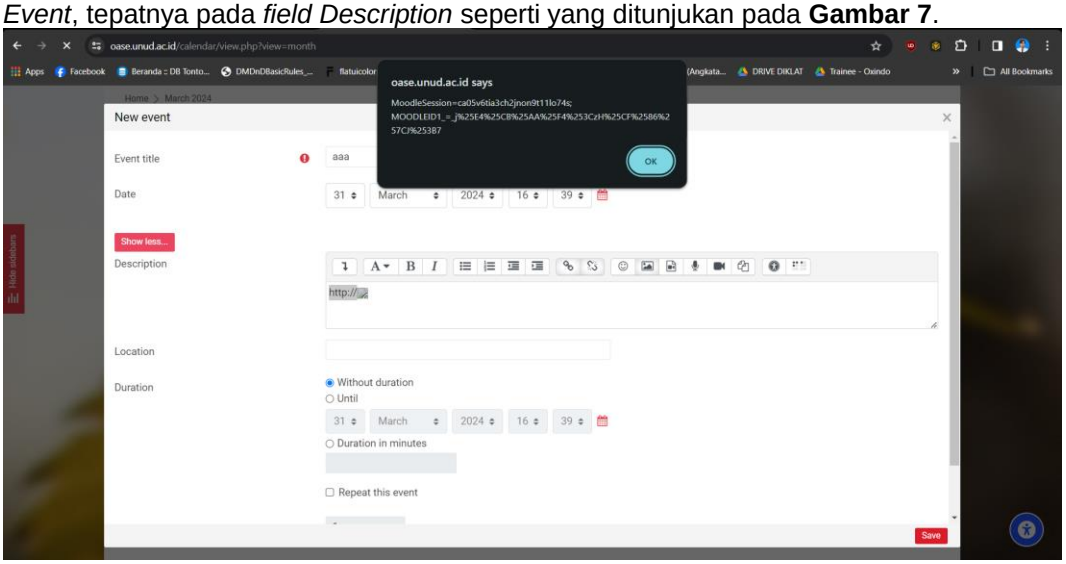
p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 5. Payload Profile

d. Fitur Calendar

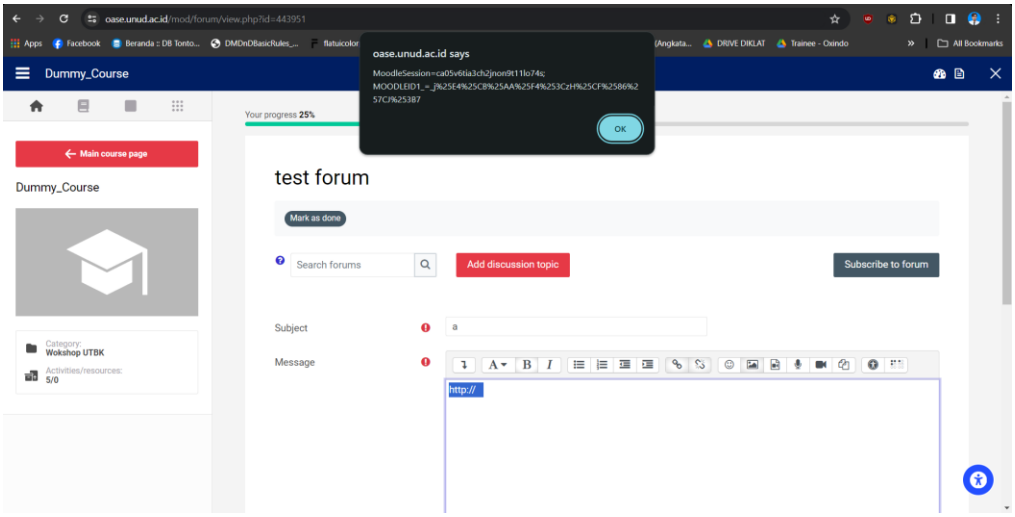
Pada fitur calendar, potensi celah keamanan ditemukan pada fungsi *Add Event*, yakni field *Event Name* dan *Description*. Dilakukan pengujian yang sama seperti pada profile, yakni menggunakan *payload* standar pengujian XSS dan *payload* yang disisipkan menggunakan *hyperlink*. Hasilnya ditemukan kerentanan serangan XSS pada fitur *Add Event*, tepatnya pada *field Description* seperti yang ditunjukan pada **Gambar 7**.



Gambar 6. Penetration Testing Pada Calendar

e. Fitur Forum

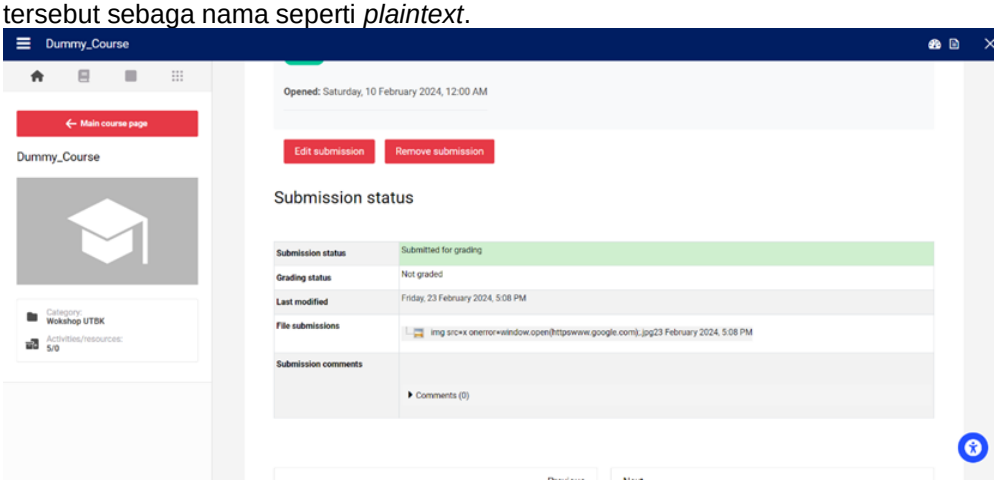
Potensi celah keamanan pada fitur forum ditemukan pada bagian *Subject* dan *Message forum*. Dimana pada kedua bagian ini terdapat *field* untuk mengisi *text* dan *script*. Setelah dilakukan pengujian menggunakan *payload* standar pengujian XSS dan *payload* yang disisipkan menggunakan *hyperlink* yang akan memunculkan *pop up* ketika terjadi error dalam load gambar, dan hasilnya ditemukan kerentanan *pada field Message* di fitur Forum seperti yang ditunjukan pada **Gambar 7.**



Gambar 7. Hasil Penetration Testing Forum

f. Fitur Assignment

Fitur Assignment sendiri memungkinan pengguna melakukan *upload file* untuk pengumpulan tugas. Potensi celah kerentanan pada fitur ini yakni pada nama *file* di *server* serta nama *file* di *local*. Pengujian dilakukan dengan menggunakan *payload* img di kedua nama *file* untuk melihat apakah *payload* akan dieksekusi oleh sistem. Hasilnya sistem tidak melakukan eksekusi *payload* dan hanya menggunakan *payload* tersebut sebaga nama seperti *plaintext*.

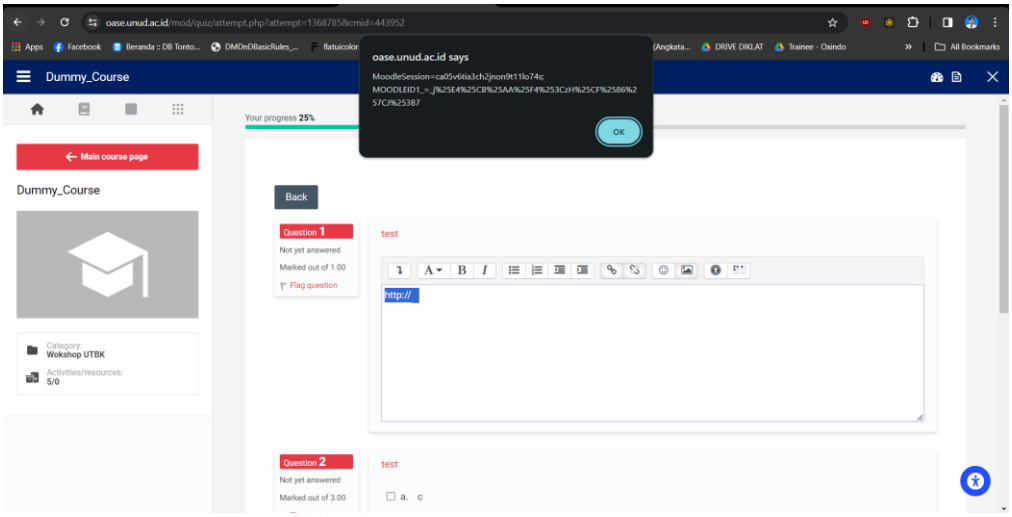


Gambar 8. Hasil Penetration Testing Assignment

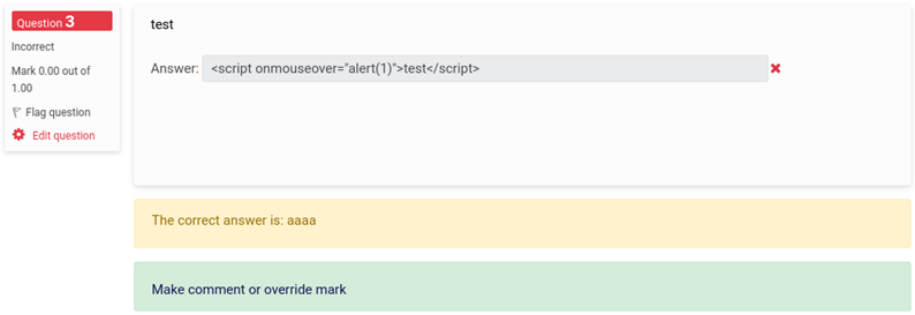
g. Fitur Quiz

Pada fitur Quiz terdapat berbagai macam jenis pertanyaan, yang akan diuji merupakan pertanyaan berbentuk Long *Answer* serta *Short Answer*. Pengujian dilakukan dengan menggunakan *payload* standar pengujian XSS dan *payload* yang disisipkan menggunakan *hyperlink*. Hasilnya seperti pada **Gambar 9** dimana pada pertanyaan *Long Answer*, *payload* pada *hyperlink* berhasil tereksekusi dan memunculkan *cookies* pengguna ketika *error* terjadi. Sedangkan pada bagain *Short Answer*, *payload* yang digunakan hanya menjadi *plaintext* tanpa tereksekusi oleh sistem seperti yang ditunjukan pada **Gambar 10.**

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 10. Hasil Penetration Testing Long Answer Quiz



Gambar 11. Hasil Penetration Testing Short Answer Quiz

3.3 Report and Evaluation

Terdapat kerentanan serangan *Cross Site Scripting* atau XSS pada hampir semua fitur yang diuji. Namun meskipun demikian, secara garis besar hanya terdapat dua fitur atau *endpoint* yang memiliki kerentanan yang ditemukan. Pertama adalah *field* pesan pada fitur Chat, pengujian lebih lanjut tidak dapat dilakukan karena fitur tersebut sudah ditiadakan oleh pemilik sistem selama proses penelitian. Dari pengujian awal, didapatkan dampak terburuk dari kerentanan ini adalah penyerang dapat mengambil *cookies* dari pengguna, sehingga akun OASE pengguna dapat diambil alih oleh penyerang.

Kedua adalah *field* message, tepatnya pada fitur *hyperlink*. Meskipun berada pada menu dan halaman yang berbeda, namun seluruh kerentanan XSS yang ditemukan pada dasarnya bersumber dari fitur *hyperlink* yang ada pada *field message*. Pada pengujian lebih lanjut, dampak yang ditimbulkan dari kerentanan ini tidak berdampak besar pada sistem, satu satunya dampak yang dihasilkan adalah *pop up* error tidak dapat ditutup oleh pengguna kecuali pengguna melakukan *refresh* halaman dimana hal ini terbilang mengganggu ketersediaan sistem. Lebih lanjut lagi, dampak yang dihasilkan hanya berlaku kepada sistem milik penyerang dan tidak berdampak kepada pengguna lain, dimana dampaknya akan hilang ketika halaman direfresh.

4. Kesimpulan

Berdasarkan hasil penelitian ini, disimpulkan terdapat beberapa potensi celah kerentanan serangan XSS pada *Website* OASE, dimana terdapat dua fitur yang kerentanannya dikonfirmasi, yaitu fitur Chat yang langsung diperbaiki oleh pengembang sistem untuk menghindari kerusakan lebih lanjut serta fitur *hyerplink* pada *field message*.

Adapun dampak yang dihasilkan tidak terlalu berdampak kepada sistem. Dimana kerentanan pada fitur *hyperlink* ini hanya berdampak kepada sisi penyerang serta tidak memiliki dampak kepada pengguna lain, serta hanya berdampak kepada ketersediaan sistem dimana hal ini akan kembali normal ketika halaman *direfresh*.

References

[1] I. Riadi, R. Umar, and T. Lestari, “Analisis Kerentanan Serangan Cross Site Scripting (XSS) pada Aplikasi Smart Payment Menggunakan Framework OWASP,” *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 5, no. 3, pp. 146–152, 2020, doi: 10.14421/jiska.2020.53-02.

[2] BSSN, “Rilis Launching Lanskap Keamanan Siber Tahun 2022,” *Bssn*, vol. 1, no. 9, p. 20, 2023.

[3] B. Ghozali, K. Kusrini, and S. Sudarmawan, “Mendeteksi Kerentanan Keamanan Aplikasi Website Menggunakan Metode Owasp (Open Web Application Security Project) Untuk Penilaian Risk Rating,” *Creat. Inf. Technol. J.*, vol. 4, no. 4, p. 264, 2019, doi: 10.24076/citec.2017v4i4.119.

[4] A. S. Hakim, T. A. Cahyanto, and H. Azizah, “Serangan cross-site scripting (XSS) berdasarkan base metric CVSS V.2,” *J. Smart Teknol.*, vol. 2, no. 1, 2020.

[5] S. Shekhar and A. Agrawal, “A survey of website security vulnerabilities,” *J. Inf. Secur. Appl.*, vol. 53, no. 102536, 2020.

[6] V. Nithya, S. Lakshmana Pandian, and C. Malarvizhi, “A survey on detection and prevention of cross-site scripting attack,” *Int. J. Secur. its Appl.*, vol. 9, no. 3, pp. 139–152, 2015, doi: 10.14257/ijsia.2015.9.3.14.

[7] OWASP, “OWASP Top Ten Project.” Accessed: Mar. 20, 2023. [Online]. Available: https://owasp.org/www-project-top-ten/