DINAMIKA REKAYASA
http://dinarek.unsoed.ac.id

Vol. 19 No. 1 (2023) Hal. 29-35
p-ISSN 1858-3075 | **e-ISSN 2527-6131**

ANALISIS CELAH KEAMANAN DAN MITIGASI WEBSITE E-LEARNING ITERA MENGGUNAKAN OWASP ZED ATTACK PROXY (ZAP)

VULNERABILITY AND MITIGATION ANALYSIS OF THE ITERA E-LEARNING WEBSITE USING OWASP ZED ATTACK PROXY (ZAP)

Ilham Firman Ashari*[1], Leonard Rizta Anugrah P[2], Nazla Andintya W[3], Siraz Tri Denira[4]

*Email: firman.ashari@if.itera.ac.id

[1,2, 3,4] Jurusan Teknik Elektro, Informatika, dan Sistem Fisis, Jurusan Teknologi Produksi dan Industri, Institut Teknologi Sumatera, Lampung Selatan.

Abstrak— Website kuliah atau *E-Learning* ITERA merupakan sistem yang mewadahi proses belajar mengajar antara dosen dan mahasiswa. Keamanan informasi dari sistem ini sangat penting mengingat banyaknya data terkait nilai ataupun materi perkuliahan yang ada pada *website e-learning* ITERA. *Default password* pada *e-learning* ITERA sendiri sudah ditetapkan. Kredensial pasti juga sudah ditetapkan setiap pengguna yaitu email ITERA. Percobaan ini bertujuan mengetahui celah keamanan menggunakan *reverse brute force attack* dengan *tools* OWASP ZAP. Adapun beberapa serangan yang teridentifikasi sebanyak 9 celah keamanan yaitu *Cross-Domain JavaScript Source File Inclusion, Incomplete or No Cache-control Header Set, Cookies without SameSite Attribute, Timestamp Disclosure – Unix, Absence of Anti-CSRF Tokens, X-Content-Type-Options Header Missing, Cookies No HttpOnly Flag, SQL Injection, .htaccess Information Leak, Absence of Anti-CSRF Tokens, Cookies No HttpOnly Flag*. Hasil pengujian membuktikan bahwa *website e-learning* ITERA berhasil diserang dengan metode *reverse brute force attack* dengan pembuktian terdapat 3 URI dengan kerentanan yang beresiko tinggi dengan serangan SQL *Injection* setelah dipindai menggunakan OWASP ZAP. Hasil mitigasi untuk risiko ini adalah dengan menggunakan *query sql* dan filter input ke database. Kerentanan Resiko *high* memerlukan penanganan secepat mungkin karena memiliki resiko yang sangat signifikan kepada sistem. Kerentanan resiko medium juga perlu ditangani secepatnya karena dapat menyebabkan berbagai data yang seharusnya tidak terlihat oleh pengguna dapat diakses. Begitu pula pada kerentanan resiko *low* perlu ditangani karena dapat memicu celah – celah keamanan lain yang lebih besar, misalnya pada *Absence of Anti CSRF Token* dapat memicu serangan berupa *brute force*.

Kata kunci — E-*learning, OWASP, brute force attack, audit, celah keamanan.*

Abstract — The college website or ITERA E-Learning is a system that accommodates the teaching and learning process between lecturers and students. Information security from this system is very important considering the large amount of data related to grades or lecture material available on the ITERA e-learning website. The default password for ITERA e-learning itself has been set. The exact credentials have also been set for each user, namely the ITERA email. This experiment aims to find security holes using a reverse brute force attack with OWASP ZAP tools. As for some of the attacks identified as many as 9 security holes, namely Cross-Domain JavaScript Source File Inclusion, Incomplete or No Cache-control Header Set, Cookies without SameSite Attribute, Timestamp Disclosure – Unix, Absence of Anti-CSRF Tokens, X-Content-Type- Options Header Missing, Cookies No HttpOnly Flag, SQL Injection, .htaccess Information Leak, Absence of Anti-CSRF Tokens, Cookies No HttpOnly Flag. The test results prove that the ITERA e-learning website was successfully attacked using the reverse brute force attack method with proof that there are 3 URIs with high-risk vulnerabilities with SQL Injection attacks after being scanned using OWASP ZAP. The result of mitigating this risk is to use sql queries and filter input to the database. High risk vulnerabilities require handling as soon as possible because it has a very significant risk to the system. Medium risk vulnerabilities also need to be addressed as soon as possible because they can cause various data that should not be visible to users to be accessed. Likewise, low-risk vulnerabilities need to be addressed because they can trigger other, bigger security holes, for example, the Absence of Anti CSRF Token can trigger attacks in the form of brute force.

Keywords — *E-learning, OWASP, brute force, audit, vulnerabilities*

I. PENDAHULUAN

Institut Teknologi Sumatera atau yang biasa dikenal ITERA adalah sebuah perguruan tinggi yang berlokasi di antara kabupaten Lampung Selatan dan kota Bandar Lampung. Tujuan didirikannya ITERA adalah untuk memajukan, mengembangkan dan menyebarluaskan ilmu pengetahuan, teknologi, seni dan humaniora, guna meningkatkan kesejahteraan masyarakat Sumatera khususnya, dan bangsa budaya Indonesia sesuai dengan dinamika perkembangan zaman. masyarakat Indonesia dan masyarakat global, dengan tetap melestarikan nilai-nilai sosial, kemanusiaan dan budaya lingkungan melalui Tridharma Perguruan Tinggi.

Salah satu cara ITERA untuk mencapai tujuannya adalah dengan sarana prasana pendukung di bidang teknologi informasi [1]. ITERA lewat Unit Pelayanan Terpadu bidang Teknologi Informasi dan Komunikasi UPT TIK banyak membuat situs layanan berbasis *website*, salah satu diantaranya *website* untuk penunjang perkuliahan yaitu *website e-learning*. *Website e-learning* ITERA memudahkan proses pembelajaran antara dosen dan mahasiswa terutama mahasiswa yang sedang menempuh Tahap Persiapan Bersama (TPB) [2][3]. *Website e-learning* ini menerapkan *profile user* yang mana setiap mahasiswa baru dapat mengaksesnya menggunakan *username (email)* dan *password*. *Password* ini umumnya bisa diubah sesuai keinginan pengguna.

Pengembangan *website e-learning* ini tentulah memerlukan keamanan khusus agar terhindar dari berbagai serangan. Semakin banyaknya celah keamanan pada suatu *website*, maka semakin mudah pula seorang penyerang untuk membajak atau *hacking website* dan mengambil ataupun mengubah data – data yang ada di dalamnya [4]. *Website e-learning* ITERA memiliki berbagai data penting terkait pembelajaran mahasiswa. Apabila penyerang dapat mengakses data – data tersebut dapat menimbulkan banyak masalah terutama terkait penilaian dan pendataan. Pada penelitian ini, penulis akan menguji dan menganalisis celah keamanan pada *website* kuliah ITERA menggunakan salah satu metode penyerangan yaitu *brute force* [5].

Serangan *brute force* adalah teknik menyerang sistem keamanan komputer dengan melakukan sebanyak mungkin percobaan terhadap semua kunci yang memungkinkan [6][7]. Serangan *brute force* digunakan untuk membobol akses *host* seperti *server, network workstation* atau kepala data yang terenkripsi [6]. Penggunaan kode sembarangan, penggunaan kata kunci yang mudah ditebak seperti nama dan nomor telepon, memungkinkan penyerang menebak kode dengan benar karena tidak aman [8]. *Brute force* bergantung pada rumitnya kode, bisa mengambil waktu berbulan – bulan bagi penyerang untuk meretasnya, terlebih jika menggunakan kriptografi yang menjadi algoritma standard [9][10]. Umumnya *brute force* tidak serumit *low technology* seperti algoritma hacking yang berkembang sekarang. Penyerang cukup menebak kombinasi nama dan kode yang cocok [11]. *Brute force* memiliki salah satu metode yaitu *reverse brute force attack*. *Reverse brute force* adalah percobaan peretasan menggunakan satu sandi yang umum. Metode ini digunakan karena sudah ada kredensial yang pasti yaitu email ITERA pengguna yang berasal dari nama depan dan nim. Selanjutnya penyerang hanya perlu menemukan *password* pengguna [12]. *Default password* pada *e-learning* ITERA sendiri sebelumnya sudah diatur sama untuk setiap akun sehingga percobaan peretasan menggunakan metode ini dinilai dapat mempermudah penulis.

Penyerangan *reverse brute force* ini memerlukan alat otomatisasi untuk mengetahui kata sandi yang cocok [13]. Penulis memilih OWASP ZAP sebagai *tools* penyerangan. *Open Web Application Security Project* (OWASP) merupakan komunitas terbuka yang mengembangkan serta memelihara aplikasi yang bisa dipercaya [14][15]. Hal ini memungkinkan untuk memberikan informasi terkait keamanan aplikasi. *Zed Attack Proxy* (ZAP) adalah sebuah aplikasi yang dapat melakukan *pentest* untuk menemukan kerentanan pada website dengan cara yang mudah. ZAP memiliki beberapa keunggulan seperti *open source, active scanner, intercept proxy, traditional and ajax spider* [16].

II. TINJAUAN PUSTAKA

A. Vulnerability

CIA atau biasa disebut *Confidentiality, Integrity, and Availability* merupakan salah satu parameter yang biasa digunakan dalam analisis kerentanan keamanan dan digunakan sebagai acuan keamanan website. Parameter ini juga menjadi tolak ukur dan acuan dalam mengevaluasi apakah keamanan informasi jaringan baik atau buruk [17].

B. *Penetration Testing*

Pengujian penetrasi adalah metode evaluasi keamanan informasi dalam sistem komputer atau jaringan dengan mengidentifikasi kerentenan. Identifikasi kerentanan keamanan, setelan *firewall*, dan *hotspot Wi-Fi*. Simulasi dan identifikasi

berlangsung di jaringan internal atau jarak jauh. Tujuannya adalah untuk menentukan dan mengetahui jenis serangan yang mungkin terjadi pada sistem dan konsekuensi yang dapat ditimbulkan dari kerentanan keamanan pada sistem atau jaringan komputer khusus. Kerentanan dalam aplikasi *server web* memungkinkan peretas untuk melakukan serangan pada sistem dan tidak mencegah pengambilalihan sistem yang diserang sepenuhnya.

C. *SQL Injection*

SQL *Injection* adalah kerentanan yang memungkinkan penyerang memengaruhi kueri SQL yang dikirim ke *database* oleh aplikasi. Fitur basis data ini memungkinkan penyerang mempengaruhi sintaks SQL, kinerja dan fleksibilitas basis data dukungan fitur, dan mempengaruhi operasi sistem operasi yang terdokumentasi ke basis data. Injeksi SQL tidak hanya mempengaruhi aplikasi web, tetapi juga semua program lain yang menggunakan pernyataan SQL. Setiap program yang menggunakan masukan dinamis dari luar (tidak dipercaya) dapat diserang oleh SQL.

D. *Cross Site Scripting (XSS)*

Cross Site Scripting dapat diartikan sebagai kerentanan yang disebabkan oleh *server* yang tidak dapat memvalidasi input pengguna. Skrip *web* adalah kerentanan yang dapat dieksploitasi yang populer. Namun, banyak penyedia layanan yang tidak menyadari kelemahan tersebut dan melakukan perubahan pada sistem yang digunakan.

E. Open Web Application Security Project (OWASP)

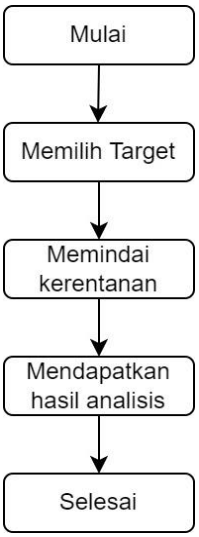
OWASP) adalah kerangka kerja sumber terbuka yang berfokus pada peningkatan keamanan perangkat lunak aplikasi. OWASP adalah organisasi yang dirancang untuk menemukan kerentanan keamanan dalam aplikasi web.

F. Open Web Application Security Project (OWASP) ZAP

OWASP ZAP (*Zed Attack Proxy*) adalah aplikasi yang digunakan untuk pengujian penetrasi untuk menemukan kerentanan atau celah keamanan pada aplikasi *web.* ZAP menawarkan pemindaian otomatis. Proyek Keamanan Aplikasi *Web* Terbuka (OWASP) adalah kerangka kerja sumber terbuka yang berfokus pada peningkatan keamanan perangkat lunak aplikasi. OWASP adalah organisasi yang dirancang untuk menemukan kerentanan keamanan dalam aplikasi *web*.

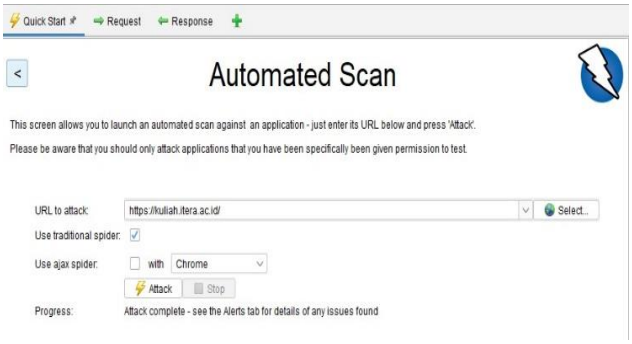
III. METODE

Berikut flowchart yang berisikan langkah – langkah audit celah keamanan pada website *e-learning* ITERA dapat dilihat pada gambar 1.



Gambar-1. Alur penyelesaian

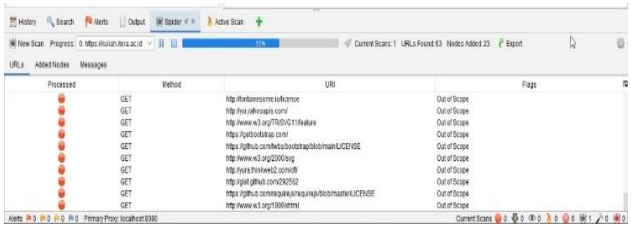
Pada percobaan ini, penulis memilih *website e-learning* kuliah ITERA yang mana pada *URL to attack* diinputkan alamat *website* yaitu https:/kuliah.itera.ac.id/. Penulis menggunakan *traditional spider* yang merupakan salah satu metode *web crawler* yang digunakan untuk menemukan dan memindai alamat URL dari *website* yang dicari celah keamanan. Lalu pilih tombol ***attack*** untuk memulai penyerangan pada *website* seperti yang terlihat pada gambar 2.



Gambar-2. Tampilan awal dari automated scan

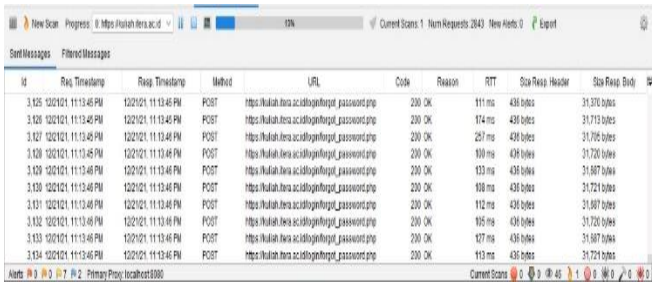
Setelah *attack*, maka disini situs tersebut akan dijelajahi menggunakan *traditional spider*. *Traditional spider* ini akan menemukan URL pada situs *e-learning* ITERA yang mana nantinya akan

diidentikasi *hyperlink* pada halaman dan ditambahkan ke daftar URL untuk dikunjungi. Lalu pilih *active scan* untuk memindai kerentanan pada *website*. Proses *scanning* dapat dilihat pada gambar 3.



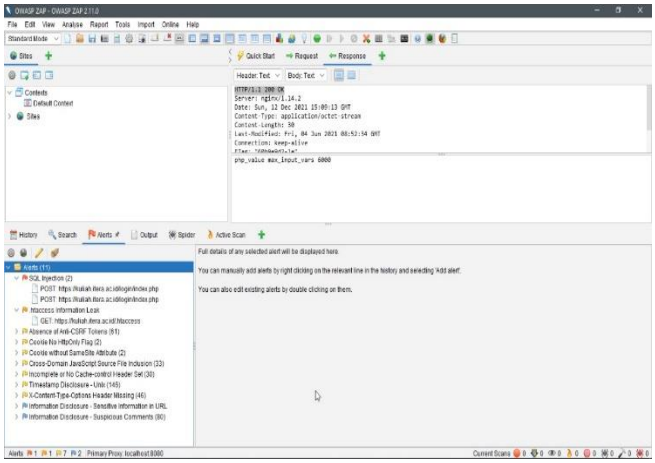
Gambar-3. Proses *scanning* dari website e-learning ITERA dengan OWASP

Selanjutnya pemindaian kerentanan pada *website e-learning* ITERA akan muncul seperti gambar dibawah. Tunggu sampai hasil menjadi 100% sehingga semua kerentanan akan muncul mulai dari *low* hingga *high*.



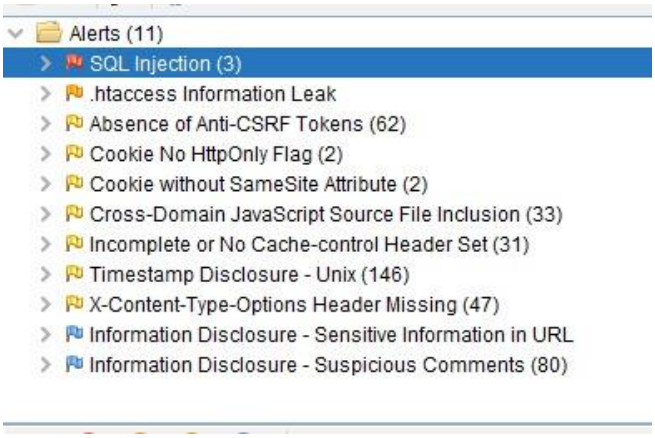
Gambar-4. Hasil scanning dari website E-Learning

Setelah melakukan penyerangan ke *website* kuliah.itera.ac.id maka didapatkan beberapa *alerts* terkait hasil analisis beberapa celah keamanan menggunakan tools OWASP ZAP. Dengan menggunakan OWASP Zap didapatkan *alert* sebanyak 11 seperti yang terlihat pada gambar 5.



Gambar-5. Tampilan secara keseluruhan celah keamanan yang didapatkan

Berikut beberapa hasil *alerts* pada penyerangan website *e-learning* ITERA yang memuat 11 celah keamanan yaitu *Cookies without SameSite Attribute, Cross-Domain JavaScript Source File Inclusion, Incomplete or No Cache-control Header Set, Timestamp Disclosure – Unix, X-Content-Type-Options Header Missing, Absence of Anti-CSRF Tokens, Cookies No HttpOnly Flag, SQL Injection, .htaccess Information Leak, Absence of Anti-CSRF Tokens, Cookies No HttpOnly Flag* [18].



Gambar-6. Detail Jumlah URI dari 11 celah keamanan

IV. HASIL DAN PEMBAHASAN

Berikut adalah hasil analisis percobaan penyerangan pada *website e-learning* ITERA. Hasil analisis dari setiap jenis serangan dapat dilihat pada table 1.

Tabel-1. Hasil analisis dari celah keamanan

No	Alert	Risk			Hasil Analisis
		High	Med	Low	
1	*SQL Injection*	✓			Hasil dari Audit menunjukkan bahwa pada *form login ke e-learning* ITERA terdapat indikasi bahwa *SQL Injection* dapat dilakukan, hal ini perlu dilakukan penanganan secepat mungkin, sehingga kemungkinan *SQL injection* ini dapat diminimalisir ataupun dihilangkan.
2	*.htaccess Information Leak*		✓		Hasil dari *Audit* menunjukkan bahwa pada file .htaccess pada *web e-learning* ITERA dapat diakses, hal ini perlu dilakukan penanganan dikarenakan file .htaccess

No	Alert	Risk			Hasil Analisis
		High	Med	Low	
					dapat berisi konfigurasi dan fitur pada *web server* yang digunakan, sehingga lebih mengamankan sisi *server side.*
3	*Absence of Anti-CSRF Tokens*			✓	Hasil dari Audit menunjukkan bahwa pada beberapa form yang ada pada *web e-learning* ITERA menunjukan tidak adanya *anti-csrf token*, sehingga hal ini dapat memicu terjadinya serangan *brute force* dikarenakan tidak adanya *process validasi csrf token* pada *form* yang ada. Maka dari itu hal ini perlu dilakukan penanganan juga untuk meminimalisir kemungkinan tersebut.
4	*Cookie No HttpOnly Flag*			✓	Hasil dari Audit menunjukkan bahwa *cookie pada web e-learning* ITERA dapat diakses melalui *javascript*, jika script yang berbahaya dapat dijalankan maka dapat memicu terjadinya *session hijacking*, maka dari itu hal ini perlu dilakukan penanganan juga untuk meminimalisir kemungkinan tersebut.
5	*Cookie without SameSite Attribute*			✓	Hasil dari Audit menunjukkan bahwa terdapat *cookie* yang diset tanpa atribut *samesite*, sehingga *cookie* tersebut dapat dikirim sebagai *cross site request*, sehingga dapat menyebabkan kemungkinan *cookie* yang disimpan dapat ada dapat dibaca oleh orang yang tidak berwenang,maka dari itu hal ini perlu dilakukan penanganan juga untuk meminimalisir kemungkinan tersebut.
6	*Cross-Domain JavaScript Source File Inclusion*			✓	Hasil dari Audit menunjukkan bahwa terdapat beberapa *script* yang dimuat dari pihak ketiga, hal ini perlu dipastikan bahwa pihak ketiga merupakan pihak yang aman.
7	*Incomplete or No Cache-control Header Set*			✓	Hasil dari Audit menunjukkan bahwa *header cache-control* belum di atur dengan benar atau hilang, maka dari itu hal ini perlu dicek untuk memastikan *browser* dapat memuat isi konten dari *cache* atau tidak.
8	*Timestamp Disclosure - Unix*			✓	Hasil dari Audit menunjukkan bahwa *timestamp* dari *server* diperlihatkan, hal ini dapat dipastikan dahulu apakah *timestamp* yang ditunjukan bersifat sensitif atau tidak
9	*X-Content-Type-Options Header Missing*			✓	Hasil dari Audit menunjukkan bahwa tidak adanya konfigurasi *header x-content-type*, sehingga hal ini dapat menyebabkan terjadinya *mime sniffing* dari *response body* pada *web*. sehingga perlu dilakukan penanganan dengan menambahkan *header x-content-type* pada *reponse header* dari *website*.

Berikut hasil mitigasi atau rekomendasi perbaikan dari jenis serangan yang diidentifikas dapat dilihat pada table 2.

Tabel-2. Rekomendasi dan mitigasi dari celah keamanan

No	Alert	Jumlah Vulnerability	Rekomendasi Perbaikan
1	*SQL Injection*	2	Lakukan validasi masukan, proses validasi masukan ini bertujuan untuk memverifikasi apakah jenis masukan yang dikirimkan oleh pengguna diperbolehkan atau tidak. Validasi masukan memastikan bahwa itu adalah jenis, panjang, format yang diterima,

No	Alert	Jumlah Vulnerability	Rekomendasi Perbaikan
			dan sebagainya. Sehingga hanya nilai yang lolos validasi yang dapat diproses. Lakukan pra-kompilasi *query* SQL, sehingga *query* dicek dahulu oleh sistem sebelum di eksekusi oleh database.
2	*.htaccess Information Leak*	1	Melakukan configurasi *file* pada *web server* untuk memastikan file *server-side* seperti *.htaccess* tidak dapat diakses
3	*Absence of Anti-CSRF Tokens*	61	Melakukan konfigurasi tambahan dengan bentuk token yang *valid* sementara ke pengiriman formulir apa pun, dimana token tersebut hanya dapat digunakan dengan sekali pakai dan tiap token harus unik, sehingga tidak ada token yang sama antara satu dengan yang lainnya.
4	*Cookie No HttpOnly Flag*	2	Memastikan bahwa *flag HttpOnly* diatur untuk semua *cookie*.
5	*Cookie without SameSite Attribute*	2	Memastikan atribut situs yang sama diatur ke *'lax'* atau idealnya *'strict'* untuk semua *cookie*
6	*Cross-Domain JavaScript Source File Inclusion*	33	Memastikan file sumber *JavaScript* dimuat hanya dari sumber tepercaya, dan sumber tidak dapat dikontrol oleh *end user aplikasi*
7	*Incomplete or No Cache-control Header Set*	30	Memastikan *header HTTP kontrol-cache* disetel dengan *no-cache, no-store, must-revalidate*
8	*Timestamp Disclosure - Unix*	146	Mengonfirmasi secara manual bahwa data stempel waktu tidak sensitif, dan bahwa data tidak dapat digabungkan untuk mengungkapkan pola yang dapat dieksploitasi
9	*X-Content-Type-Options Header Missing*	46	Menambahkan *header x-content-type* pada *reponse header* dari *website*. Memastikan bahwa *end user* menggunakan *browser web* yang terbaru sehingga sesuai standar yang tidak melakukan *sniffing MIME*, atau yang dapat diarahkan oleh aplikasi *web/server web* untuk tidak melakukan *sniffing MIME*.

V. PENUTUP

A. Kesimpulan

Berdasarkan audit celah keamanan pada website yang telah dilakukan, *website e-learning* ITERA memiliki berbagai jenis kerentanan yang mana diantaranya berpotensi memiliki resiko yang *low*, *medium*, dan *high*. Resiko *high* memerlukan penanganan secepat mungkin karena memiliki resiko yang sangat signifikan kepada sistem. Kerentanan resiko *medium* juga perlu ditangani secepatnya karena dapat menyebabkan berbagai data yang seharusnya tidak terlihat oleh pengguna dapat diakses. Begitu pula pada kerentanan resiko *low* perlu ditangani karena dapat memicu celah – celah keamanan lain yang lebih besar resikonya, misalnya pada *Absence of Anti CSRF Token* dapat memicu serangan berupa *brute force*. Sehingga diperlukannya penanganan pada celah keamanan yang ditemui.

B. Saran

Berdasarkan percobaan yang telah dilakukan terdapat beberapa saran sebagai berikut :

1. Percobaan selanjutnya dapat menggunakan tools selain OWASP yang memungkinkan adanya celah keamanan lain pada *e-learning* ITERA seperti *Information Systems Security Assessment Framework*.
2. Percobaan yang dapat dilakukan selanjutnya dalam audit celah keamanan dapat menggunakan teknik lain seperti *SQL Injection*, karena celah pada teknik tersebut merupakan celah yang memiliki kerentanan yang tinggi.

DAFTAR PUSTAKA

[1] I. F. Ashari, "Analysis and Implementation of Augmented Reality Using Markerless and A-Star Algorithm (Case Study: Gedung Kuliah Umum ITERA)," *Comput. Eng. Appl. J.*, vol. 11, no. 3, pp. 177–190, doi: 10.18495/comengapp.v11i3.414, 2022.

[2] lham F. Ashari, A. Afriansyah, A. Setiawan, and E. D. Nugroho, "Teaching Science with BeSmart E-Learning Technology at SMA Negeri 1 Tanjung Sari, South Lampung," ABDIMAS J. Pengabdi. Masy., vol. 5, no. 2, 2022.

[3] I. F. Ashari, M. Idris, and M. A. Nasrulah, "Analysis of Combination of Parking System with Face Recognition and QR Code using Histogram of Oriented Gradient Method," *IT J. Res. Dev.*, vol. 7, no. 1, pp. 94–110, doi: 10.25299/itjrd.2022.9958, 2022.

[4] Andria, "Analisis Celah Keamanan Website Menggunakan Tools WEBPWN3R di Kali Linux," *Gener. J.*, vol. 4, no. 2, pp. 69–76, 2020.

[5] H. S. Pratita, "Analisa Brute Force Attack menggunakan Scanning Aplikasi pada HTTP Attack," *2017*, no. 672010194, 2017.

[6] Y. Mulyanto, H. Herfandi, and R. Candra Kirana, "Analisis Keamanan Wireless Local Area Network (Wlan) Terhadap Serangan Brute Force Dengan Metode Penetration Testing (Studi Kasus:Rs H.Lmanambai Abdulkadir)," *J. Inform. Teknol. dan Sains*, vol. 4, no. 1, pp. 26–35, doi: 10.51401/jinteks.v4i1.1528, 2022.

[7] J. Simarmata, D. Sasongko, and I. F. Ashari, *Sistem Keamanan Data*, vol. 5, no. 3. 2022.

[8] I. Gunawan, "Modifikasi Keamanan File dengan Algoritma Hill Cipher Untuk Mengantisipasi Dari Serangan Brute Force," *TECHSI - J. Tek. Inform.*, vol. 11, no. 2, p. 237, doi: 10.29103/techsi.v11i2.1272, 2019.

[9] I. F. Ashari, A. W. Bhagaskara, J. M. Cakrawarty, and P. R. Winata, "Image Steganography Analysis Using GOST Algorithm and PRNG Based on LSB," *Techno.Com*, vol. 21, no. 3, pp. 700–713, doi: 10.33633/tc.v21i3.6331, 2022.

[10] I. F. Ashari, "The Evaluation of Audio Steganography to Embed Image Files Using Encryption and Snappy Compression," *Indones. J. Comput. Sci.*, vol. 11, no. 2, pp. 318–336, 2022.

[11] Syaifuddin, D. Risqiwati, and E. A. Irawan, "Realtime Pencegahan Serangan Brute Force dan DDOS Pada Ubuntu Server," *Techno.Com*, vol. 17, no. 4, pp. 347–354, doi: 10.33633/tc.v17i4.1766, 2022.

[12] R. Komalasari, "Kesadaran Akan Keamanan Penggunaan Username Dan Password," *Tematik*, vol. 5, no. 2, pp. 56–67, 2018, doi: 10.38204/tematik.v5i2.265.

[13] Y. Indarta, F. Ranuhardja, and I. F. Ashari, *Keamanan Siber Tantangan di Era Revolusi Industri 4.0*. 2022.

[14] M. Yunus, "Analisis Kerentanan Aplikasi Berbasis Web Menggunakan Kombinasi Security Tools Project Berdasarkan Framework Owasp Versi 4," *J. Ilm. Inform. Komput.*, vol. 24, no. 1, pp. 37–48, doi: 10.35760/ik.2019.v24i1.1988, 2019.

[15] I. F. Ashari, M. Alfarizi, M. N. K, and M. A. H, "Vulnerability Analysis and Proven On The neonime. co Website Using OWASP ZAP 4 and XSpear," *J. Teknol. Komput. dan Sist. Inf.*, vol. 5, no. 2, pp. 75–81, 2022.

[16] 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, doi: 10.24076/citec.2017v4i4.119, 2019.

[17] I. F. Ashari, V. Oktariana, R. G. Sadewo, and S. Damanhuri, "Analysis of Cross Site Request Forgery (CSRF) Attacks on West Lampung Regency Websites Using OWASP ZAP Tools," *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 11, no. 2, pp. 276–281, 2022.

[18] T. Hidayah and H. Saptono, "Jurnal Informatika Terpadu PENERAPAN HIGH AVAILABILITY WEB SERVER MENGGUNAKAN NGINX DAN MODSECURITY," *J. Inform. Terpadu*, vol. 3, no. 2, pp. 95–102, 2017.