Jurnal ICT : Information Communication & Technology
Vol. 20, N0.1, Juli 2021, pp. xx-xx
p-ISSN: 2302-0261, e-ISSN: 2303-3363, DOI:

PENERAPAN SISTEM KEAMANAN WEBSITE MENGGUNAKAN WEB APPLICATION FIREWALL DENGAN FRAMEWORK OPEN WEB APPLICATION SECURITY PROJECT

Muhammad Dandi Permana[1*], Syahril Rizal[2*], Febriyanti Panjaitan[3*]
[1,2,3]Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Bina Darma, Indonesia
Email: [1]191420058@student.binadarma.ac.id, [2]syahril.rizal@binadarma.ac.id , [3]suryayusra@binadarma.ac.id, [4]febriyanti_panjaitan@binadarma.ac.id

INFORMASI ARTIKEL

Histori artikel:

Naskah masuk, 23 September 2019
Direvisi, 23 September 2019
Diiterima, 23 September 2019

Kata Kunci:

Website,
WAF,
OWASP

ABSTRAK

Abstract- *Websites are one of the most popular forms of promotional media today. The website is used to provide information to customers. In addition, the data on the website must be protected from unwanted things such as XSS attacks. In order to maintain the security of the BOOM STORE website from this attack, the author has made improvements to website security. This research adopts Web Application Firewall (WAF) with ModeSecurity and ZAP penetration test tool to improve website security. Using the experimental method, WAF is applied as web protection by analysis and testing. Results show WAF ModeSecurity's success in stopping XSS attacks, ensuring BOOM STORE security.*

Abstrak- Website adalah salah satu bentuk media promosi paling populer saat ini. Website digunakan untuk memberikan informasi kepada pelanggan. Selain itu, data-data pada situs web haruslah dijaga dari hal hal-hal yang tidak diinginkan seperti XSS Attack. Demi menjaga keamanan website BOOM STORE dari serangan itu maka penulis membuat peningkatan keamanan website. Penelitian ini mengadopsi Web Application Firewall (WAF) dengan ModeSecurity dan alat tes penetrasi ZAP untuk meningkatkan keamanan website. Menggunakan metode eksperimen, WAF diterapkan sebagai proteksi web dengan analisis dan uji coba. Hasil menunjukkan keberhasilan WAF ModeSecurity dalam menghentikan serangan XSS, memastikan keamanan BOOM STORE

Penulis Korespondensi:

Muhammad Dandi Permana
Program Studi Teknik Informatika,
Fakultas Sains dan Teknologi Universitas Bina Darma
Jalan Jendral A.Yani No.3 Plaju Palembang 30264, Indoensia
Email: 191420058@student.binadarma.ac.id

Jurnal ICT : Information Communication & Technology
Vol. 20, N0.1, Juli 2021, pp. xx-xx
p-ISSN: 2302-0261, e-ISSN: 2303-3363, DOI:

1. Pendahuluan

Kemajuan teknologi dianggap telah berkembang jauh dalam periode yang selalu berubah ini. Semuanya sangat bergantung pada layanan Internet. Mulailah dengan tugas-tugas sederhana seperti menonton acara hiburan, membaca berita, atau memesan makanan dari aplikasi internet [1]. Internet menyederhanakan semua aspek keberadaan. Dengan hadirnya internet, semua pekerjaan yang ada dapat diselesaikan dengan lebih cepat, misalnya saat pandemi Covid-19. Wabah Covid-19 mendorong orang untuk mengalihkan semua operasi mereka secara online. Orang-orang tidak diizinkan keluar rumah untuk mengurangi penyebaran virus. Orang lebih cenderung membuka halaman web dan sumber daya daring lainnya sebagai akibat dari efek ini. Karena sulit bagi orang untuk meninggalkan rumah selama pandemi, lalu lintas ke setiap situs web melonjak [2].

Aplikasi web kini menjadi bagian dari masyarakat saat ini karena seluruh kebutuhan diperoleh melalui aplikasi berbasis web dan aplikasi seluler, aplikasi web kini telah menjadi bagian mendasar dari kehidupan sehari-hari. [3]. Hal ini tentunya memudahkan untuk mendapatkan informasi dari web ini. Di antara banyak manfaat aplikasi web di internet saat ini adalah kelemahan keamanan yang membuatnya rentan terhadap serangan dari pihak yang tidak bertanggung jawab. Pentingnya keamanan dalam aplikasi web tidak dapat dilebih-lebihkan. Keamanan aplikasi web adalah fitur penting yang harus diterapkan. [4].

2. Tinjauan Pustaka

2.1 Keamanan Website

Keamanan situs web adalah bagian dari keamanan informasi yang diterapkan pada situs web. Tujuan keamanan situs web adalah untuk melindungi informasi dan data pada website [5]. Semakin maju suatu teknologi ditambah dengan peningkatan nilai informasi, semakin besar kemungkinan munculnya jenis kejahatan baru. Cybercrime adalah kejahatan yang membahayakan Internet atau menggunakan Internet untuk melakukan kejahatan [6]. Cybercrime adalah kejahatan yang dilakukan oleh seseorang yang menggunakan layanan online ilegal, teknologi komputer dan internet lintas batas negara, menyebabkan kerugian dan sulit dibuktikan secara hukum. Cybercrime adalah ancaman yang jelas terhadap keamanan situs web. [7].

2.2 WAF (Web Aplication Firewall)

WAF adalah aplikasi berbasis web yang memfilter, memantau, dan memblokir lalu lintas melalui Hyper Text Transfer Protocol (HTTP). Secara umum, WAF berbeda dengan firewall. Kemampuan untuk menyaring serangan yang berasal dari berbagai macam konten yang mengarah ke aplikasi online [8]. Karena WAF dapat memantau lalu lintas Hyper Text Transfer Protocol (HTTP), itu dapat mencegah serangan seperti injeksi SQL, cross site scripting (XSS), penyertaan file, dan kesalahan penginstalan. [9].

2.3 OWASP (Open Web Application Security Project)

Open Web Application Security Project (OWASP) berkaitan dengan perlindungan aplikasi online dari kerentanan serangan. Salah satu proyek yang bekerja untuk membakukan sepuluh kelemahan keamanan dalam aplikasi web adalah OWASP Top. Pengembang menambal kerentanan keamanan sebelum aplikasi web memasuki proses produksi dengan membaca whitepaper OWASP Top 10. Berikut adalah 10 celah keamanan berdasarkan dokumen OWASP [10] :

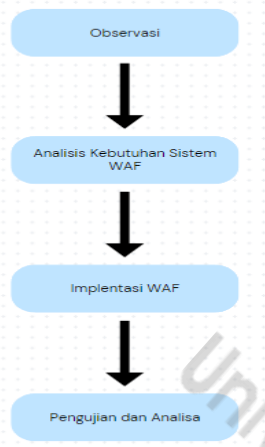
1. *Injection*
2. *Broken Authentification*
3. *Sensitive Data Exposure*
4. *XML External Entities (XXE)*
5. *Broken Acces Control*
6. *Security Misconfiguration*
7. *Cross-Site Scripting XSS*
8. *Insecure Deserialization*
9. *Components with Knows Vurnerabilities*
10. *Insufficient Logging & Monitoring*

2.4 ModSecurity

Modsecurity adalah solusi pertahanan atau keamanan berbasis open source yang menampilkan Firewall solusi Web lintas platform. Modsecurity membela aplikasi web dari berbagai jenis serangan kerentanan keamanan dan dapat memantau lalu lintas HTTP(s) secara real time [11].

3. Metode Penelitian

Penelitian ini akan menggunakan metode eksperimen dengan mengimplementasikan Web Application Firewall (WAF) sebagai sistem proteksi berbasis web, kemudian proses analisis dan ujicoba untuk memperoleh saran yang akurat dalam implementasi firewall. Adapun dalam penelitian ini terdapat 4 tahapan yaitu sebagai berikut:



Gambar 1. Tahapan Metode Penelitian

3.1 Observasi

Tahap observasi dilakukan pada web server Boom Store. Keamanan aplikasi web diterapkan untuk melindungi dari ancaman dengan membuat firewall yang terhubung dan terletak di satu jaringan server aplikasi web.

3.2 Alat dan Bahan Sistem WAF

A. Web Appliaction Firewall

WAF merupakan inti dari penelitian ini, yang akan diimplementasikan untuk melindungi aplikasi web dari serangan siber. Pilihan WAF yang kompatibel dengan Framework OWASP, seperti mod_security untuk server Apache atau NAXSI untuk server Nginx, akan menjadi alat utama dalam penerapan sistem keamanan.

B. Server Web

Server web adalah infrastruktur yang akan digunakan untuk meng-host aplikasi web yang akan diamankan. Penggunaan server seperti Apache atau Nginx akan diperlukan untuk mengaktifkan dan mengintegrasikan WAF dengan aplikasi web.

C. Framework OWASP

Framework OWASP menyediakan seperangkat aturan dan panduan keamanan yang telah terbukti efektif dalam melindungi aplikasi web dari berbagai serangan. Penggunaan OWASP sebagai panduan utama dalam mengonfigurasi WAF dan mengidentifikasi kerentanan adalah komponen integral dari penelitian ini.

D. Perangkat Lunak Uji Penetrasi

Alat uji penetrasi seperti Burp Suite, OWASP ZAP, atau Nikto akan digunakan untuk melakukan uji penetrasi terhadap aplikasi web setelah implementasi WAF. Alat-alat ini membantu mengidentifikasi kerentanan yang mungkin terlewatkan selama konfigurasi dan mengevaluasi efektivitas perlindungan yang diberikan oleh WAF.

E. Data Uji Coba

penulis memerlukan dataset yang mewakili lalu lintas yang berpotensi masuk ke aplikasi web. Data uji coba ini dapat berupa permintaan HTTP dan respons yang mencakup berbagai jenis serangan dan situasi yang mungkin terjadi di lingkungan produksi.

F. Dokumentasi OWASP

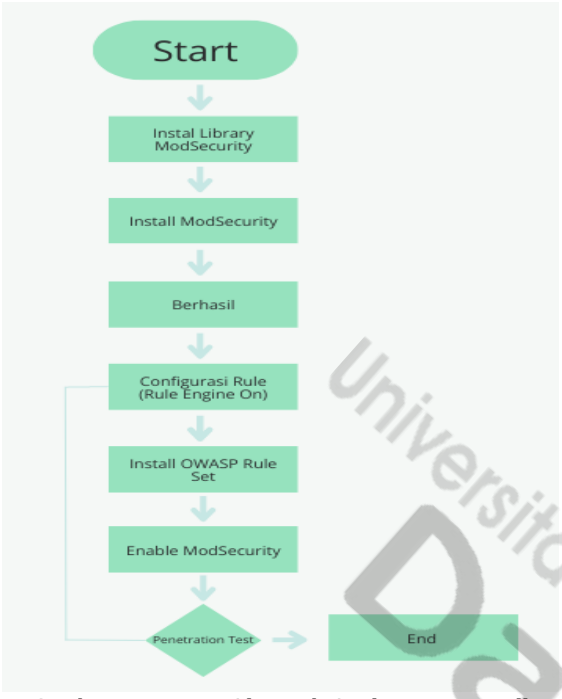
Dokumentasi dari OWASP akan menjadi sumber informasi utama untuk mengonfigurasi aturan-aturan WAF dan memahami kerentanan yang perlu diatasi. Ini termasuk panduan dan dokumentasi terbaru tentang berbagai jenis serangan dan cara-cara perlindungan.

G. Lingkungan Pengujian Terpisah

Lingkungan pengujian terpisah akan membantu penulis untuk menguji penerapan WAF tanpa risiko merusak aplikasi web di lingkungan produksi. penulis dapat menggunakan lingkungan virtual atau platform cloud untuk membuat lingkungan pengujian terisolasi.

3.3 Implementasi WAF

Dalam upaya menghadapi ancaman siber yang semakin kompleks dan bertambahnya kerentanan dalam lingkungan yang serba online ini, implementasi sistem keamanan website menjadi suatu keharusan. Penerapan Web Application Firewall (WAF) dengan dukungan Framework OWASP (Open Web Application Security Project) telah menjadi solusi yang sangat efektif untuk melindungi aplikasi web dari serangan berbahaya dan menjaga integritas serta ketersediaan data. Maka dari itu penulis akan membahas langkah-langkah dalam penerapan sistem keamanan ini, yang meliputi proses dari awal hingga akhir. Proses Implementasi Keamanan menggunakan Web Application Firewall terdiri dari bebarapa tahapan sebagai berikut:



Gambar 2. Diagram Alur Web Application Firewall

A. Implementasi Modsecurity

Berikut adalah tahapan – tahapan dalam implementasi Web Applications Firewall (WAF) menggunakan Modsecurity pada website Boom Store:

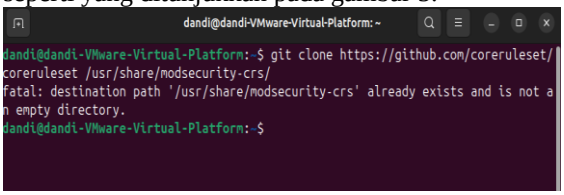
Tahap pertama adalah instalasi Web Applications Firewall (WAF) pada website admin Boom Store dengan menggunakan perintah sudo apt-get install libapache2- mod- security-2 seperti yang ditunjukkan pada gambar 3.

Gambar 3. Instalasi Web Applications Firewall (WAF)

Tahap kedua setelah instalasi selesai adalah rename file modsecurity.conf-recommended menjadi modsecurity.conf yang berada pada folder /etc/modsecurity/ lalu restart apache2 seperti telihat pada gambar 4.

Gambar 4. Ubah nama modsecurity.conf

Tahap ketiga unduh file rules pada git clone https://github.com/coreruleset/coreruleset dan diletakan dii folder /usr/share/modsecurity-crs/ seperti yang ditunjukkan pada gambar 5.



Gambar 5. Isi folder /usr/share/modsecurity-crs/

Tahap Keempat adalah ubah di sudo nano /etc/apache2/sites-available/000-default.conf untuk menambahkan file rule menjadi seperti pada gambar 6.

Gambar 6. Include konfigurasi modsecurity

Tahap kelima pada implemntasi Web Applications Firewall (WAF) adalah dengan restart apache agar module security terupdate dengan konfigurasi yang baru.

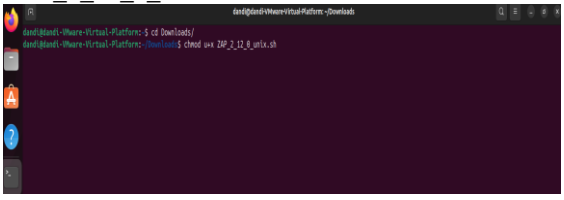
B. Instalasi OWASP ZAP

Selanjutnya adalah tahap instalasi OWASP ZAP pada Ubuntu. Berikut adalah tahapan – tahapan dalam instalasi untuk pengujian pada website Boom Store:

Tahap pertama dengan mendownload OWASP Zap pada situs resminya yaitu https://www.zaproxy.org/download/ dan hasil downloadnya akan tersimpan di cd download.

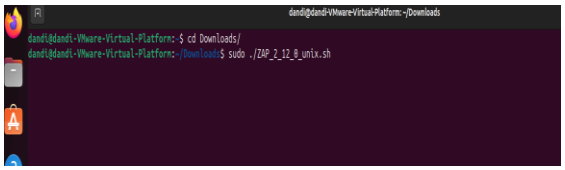
Gambar 7. Unduh ZAP

Tahap kedua adalah buka terminal dan Pergi ke cd downloads dan jalankan chmod u+x ZAP_2_12_0_unix.sh



Gambar 8. Instalasi OWASP ZAP

Tahap ketiga jalankan menggunakan sudo ./ZAP_2_12_0_unix.sh, maka otomatias akan terinstal.



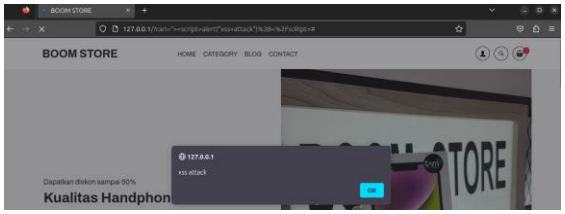
Gamabar 9. Menjalankan OWASP ZAP di ubuntu

3.4 Pengujian Sistem Keamanan

Pada Pengujian WAF (Web Application) terdapat 2 Tahap pengujian yaitu menggunakan Modsecurity sebagai keamananya dan OWASP ZAP sebagai Web Application Penetration Testing Tool. Pada pengujian menggunakan OWASP Modsecurity tedapat 2 skenario yaitu Pertama pengujian pada website Boom Store dengan status ModSecurity tidak aktif dengan serangan Cross Site Scripting (XSS) Pada skenario kedua pengujian dilakukan dengan status ModSecurity aktif dengan serangan yang sama. Pengujian ini bertujuan untuk memastikan berjalan atau tidaknya WAF (Web Application Firewall) yang sudah di setting. Setelah selesai melakukan pengujian WAF menggunakan Modsecurity, penulis akan melakukan pengujian keamanan (penetration testing) terhadap website Boom Store.

A. Percobaan XSS

Berikut merupakan tampilan sebelum menggunakan filter Web Application Firewall (WAF). Dalam contoh tersebut dapat dibuktikan bahwa Cross Site Scripting (XSS) bisa dilakukan terhadap server dengan menggunakan exploit yang dilakukan terhadap input form yang sudah disediakan.



Gambar 10. Berhasil melakukan percobaan XSS attack

Berikut merupakan tampilan sesudah menggunakan implementasi filtering Web Application Firewall (WAF). Setiap request yang dilakukan oleh user dan terdeteksi sebagai sebuah serangan maka akan dialihkan ke halaman berikut :

Gambar 11. Akses ditolak oleh WAF

Berikut merupakan hasil log yang terekam oleh ModSecurity sebagai WAF ketika attacker melakukan exploitasi dengan Cross Site Scripting.

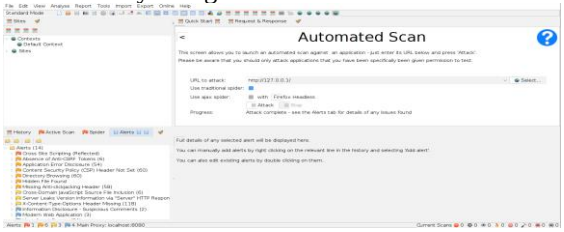
Message: Warning. detected XSS using libinjection. [file "/usr/share/modsecurity-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf"] [line "97"] [id "941100"] [msg "XSS Attack Detected via libinjection"] [data "Matched Data: XSS data found within ARGS:cari: \x22><scrIpt>alert(\x22xss attack\x22);</scRipt>"] [severity "CRITICAL"] [ver "OWASP_CRS/4.0.0-rc1"] [tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-xss"] [tag "paranoia-level/1"] [tag "OWASP_CRS"] [tag "capec/1000/152/242"]

Pesan log menunjukkan tingkat peringatan atau tindakan yang diambil oleh ModSecurity. Pada contoh ini, pesan log menampilkan tingkat peringatan "Warning", yang menunjukkan bahwa serangan XSS telah terdeteksi. Karena adanya pesan log yang berisi detected XSS using libinjection, bagian ini menyatakan bahwa serangan XSS (Cross-Site Scripting) telah terdeteksi menggunakan modul libinjection. Libinjection adalah sebuah pustaka C yang dapat mendeteksi dan mencegah serangan injeksi SQL dan XSS.

B. Penetration Testing Tool menggunakan OWASP ZAP

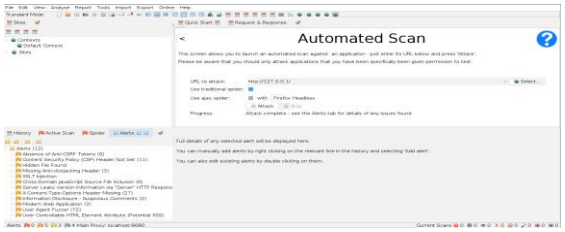
Setelah melakukan pengujian dengan Modsecurity agar lebih terlihat hasilnya, maka selanjutnya dilakukan test attack menggunakan OWASP ZAP dengan memasukan URL yang akan dilakukan pengujian. OWASP ZAP membantu Anda secara otomatis menemukan celah keamanan di aplikasi web selama pengembangan dan pengujian aplikasi. Ini juga merupakan alat yang hebat untuk penguji berpengalaman yang terbiasa dengan pengujian keamanan manual.

Pertama-tama penulis akan melakukan pengujian website Boom Store menggunakan OWASP ZAP. Ini adalah hasil tampa menggunakan Modcsecurity sebagai keamanan website.



Gambar 12. Pengujian menggunakan OWASP ZAP

hasil pengujian setelah penulis melakukan penerapan menggunakan Modsecurity sebagai keamanan website.



Gambar 13. Hasil pengujian setelah menerapkan Modsecurity pada website

Disini penulis bisa melihat perbedaan hasil dari menggunakan Modsecurity dan hasil yang tidak menggunakan modsecurity sebagai keamanan website. Terdapat beberapa perbedaan pada bagian alert nya, yaitu sebagai berikut;

1. Penggunaan ModSecurity mencegah serangan Cross-Site Scripting (XSS) dengan menghindari munculnya alert XSS pada halaman web.
2. ModSecurity mencegah Appliacation Error Disclosure yang mengungkapkan informasi sensitif pada halaman web.
3. ModSecurity melindungi website dari Directory Browsing yang dapat membocorkan daftar isi direktori.

Setelah penulis mengetahui beberapa perbedaan menggunakan Modsecurity dan yang tidak menggunakan Modsecurity pada OWASP ZAP, maka selanjutnya penulis akan Automated Scan secara langsung link link yang dianggap berbahaya dan bisa menyerangan website Boom Store. Berikut adalah hasil dari Automated Scan OWASP ZAP;

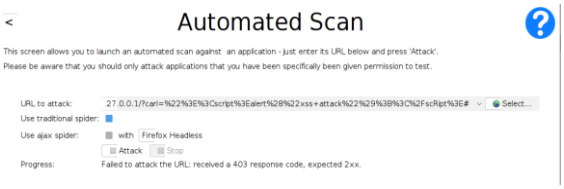
Disini penulis akan mencoba XSS attack tanpa menggunakan modsecurity, apakah masih bisa di attack menggunakan Automated Scan OWASP ZAP. Hasilnya adalah masih bisa diattack menggunakan XSS attack, keterangannya sebagai berikut;

Progress: Actively scanning (attacking) the URLs discovered by the spider(s).

Gambar 14. Masih bisa di attack

Percobaan selanjutnya penulis akan mencoba XSS attack menggunakan modsecurity, apakah masih bisa di attack menggunakan Automated Scan OWASP ZAP. Hasilnya adalah tidak bisa diattack menggunakan XSS attack. keterangannya sebagai berikut ;

Progress: Failed to attack the URL: received a 403 response code, expected 2xx.



Gambar 15. Test attack menggunakan OWASPZAP

4. Hasil dan Pembahasan

Analisis hasil penelitian dalam konteks penerapan sistem keamanan website menggunakan WAF (Web Application Firewall) dengan framework OWASP (Open Web Application Security Project) menjadi inti yang sangat penting dalam laporan penelitian ini. Dalam bagian ini, penulis akan menyajikan data yang relevan, termasuk jenis kerentanan yang teridentifikasi dan serangan yang berhasil dicegah oleh WAF.

A. Analisis Hasil Serangan XSS Pada Website

Setelah melakukan implementasi dan pengujian, hasil yang diperoleh selama implementasi dan pengujian akan dianalisis. Seperti terlihat pada Gambar 3.11 (No. halaman 15), hasil pengujian menunjukkan bahwa implementasi OWASP ModSecurity pada website berhasil mengatasi serangan XSS terhadap website. Serangan terhadap situs web dapat dideteksi menggunakan keamanan ModSecurity dengan mengidentifikasi jenis serangan XSS berdasarkan aturan keamanan ModSecurity dasar, Rule Set yang berisi aturan untuk memfilter permintaan dari requet klien untuk menyerang berdasarkan parameter yang disediakan pengguna yang sesuai dengan aturan keamanan ModSecurity.

Rule Set serangan XSS jenis ini berisi aturan yang terkait dengan kemungkinan serangan XSS, seperti karakter dan perintah yang digunakan dalam jenis serangan ini. Jadi ketika Penulis atau penyerang menyerang server web dengan jenis serangan ini, modsecurity akan merespons dengan 403. Perintah SecRule dalam aturan modsecurity digunakan untuk membuat aturan yang argumennya berdasarkan parameter yang akan di filter berdasarkan serangan XSS. Dalam serangan XSS. Log modsecurity dibuat berdasarkan perintah modsecurity SecRule menggunakan perintah 'log' dan perintah 'nolog' jika penulis tidak ingin mencatatnya di log modsecurity. Deteksi serangan yang tercatat di log Modsecurity memberikan informasi berdasarkan jenis serangan yang menyerang server web. Informasi dalam log server dapat berupa pesan tentang jenis serangan atau alamat IP penyerang. Semua informasi ini didasarkan pada konfigurasi Set Aturan Modsecurity.

B. Analisis Hasil Penetration menggunakan OWASP ZAP

Setelah melakukan implementasi dan uji coba maka diperoleh sebuah hasil dari implementasi dan uji coba tersebut yang akan dianalisa. Seperti pada gambar 3.16 (No. halaman 20) hasil dari uji coba itu menemukan bahwa implementasi menggunakan OWASP ModSecurity sekaligus dilakukannya Penetration Testing tool menggunakan OWASP ZAP pada website maka hasil penetration Tool nya gagal karena xss tidak terdeteksi pada website. Selain dari penetration tool yang gagal, ada beberapa hal yang perlu diperhatikan, antara lain sebagai berikut;

1. Identifikasi Celah Serangan

Seperti pada gambar 3.13 (No. halaman 18) terdapat bebarapa tingkat keparahan yang berhasil di didapat oleh OWASP ZAP, yaitu sebagai berikut;

No	Celah keamanan	High	Medium	Low	Informational
1	Cross Site Scripting	✓			
2	Absence of Anti_CSRF Tokens		✓		
3	Application Error Disclosure		✓		
4	Content Security Policy (CSP)		✓		
5	Directory Browsing		✓		
6	Hidden File Found		✓		
7	Missing Anti-clickjacking Header		✓		
8	Cross-Domain JavaScript Source file Inclusion			✓	
9	Server Leaks Version Information via"Server"			✓	
10	Missing Anti-clickjacking Header			✓	
11	Information Disclosure – Surpocious Comments				✓
12	Modern Web Appliaction				✓
13	User Agent Fuzzer				✓
14	User Controllable HTML Element Attribute				✓

Tabel 4.1 Identifikasi Celah Serangan

2. Perbandingan Pengujian

Terdapat perbedaan sebelum dan sesudah implementasi WAF antara lain seperti tabel berikut;

No	Serangan	Terdeteksi	Tidak terdeteksi
1	Cross site scripting (Reflected)	✓	
2	Appliacation Error Disclosure	✓	
3	Directory Browsing	✓	

Tabel 4.2 Sebelum menggunakan WAF

No	Serangan	Terdeteksi	Tidak terdeteksi
1	Cross site scripting (Reflected)		✓
2	Appliacation Error Disclosure		✓
3	Directory Browsing		✓

Tabel 4.3 Sesudah menggunakan WAF

Analisis Hasil Penetration Testing Tool menggunakan OWASP ZAP dalam penelitian ini mengambil inti dari pengujian penetrasi menggunakan OWASP ZAP dan memberikan wawasan tentang efektivitas sistem keamanan website yang diperkuat dengan WAF. Hasil analisis ini akan membantu dalam mengenali area-area yang perlu diperkuat, memperbaiki kerentanan, dan memastikan bahwa sistem keamanan dapat memberikan perlindungan yang lebih baik terhadap ancaman-ancaman keamanan aplikasi web.

5. Kesimpulan

ModSecurity sebagai Web Application Firewall (WAF) meningkatkan keamanan website dengan mencegah serangan umum pada aplikasi web. Pengujian dengan OWASP ZAP menunjukkan ModSecurity berhasil melindungi website dari serangan yang diidentifikasi. Meskipun tidak ada keamanan mutlak, penggunaan ModSecurity dan pengujian penetrasi seperti OWASP ZAP memberikan lapisan pertahanan tambahan yang kuat untuk melindungi aplikasi web dan meningkatkan keamanan keseluruhan.

Ucapan Terima kasih

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Tuhan Yang Maha Esa, kedua orang tua, pembimbing dan teman-teman atas dukungan, kasih sayang dan bantuannya selama penelitian ini..

Daftar Pustaka

[1] M. Pandemi and R. Fadhillah, "Penggunaan Teknologi dan Internet sebagai Media," 2020.

[2] A. R. Kelrey and A. Muzaki, "Pengaruh Ethical Hacking Bagi Keamanan Data Perusahaan," *Cyber Secur. dan Forensik Digit.*, vol. 2, no. 2, pp. 77–81, 2019, doi: 10.14421/csecurity.2019.2.2.1625.

[3] D. Maharani, F. Helmiah, and N. Rahmadani, "Penyuluhan Manfaat Menggunakan Internet dan Website Pada Masa Pandemi Covid-19," *Abdiformatika J. Pengabdi. Masy. Inform.*, vol. 1, no. 1, pp. 1–7, 2021, doi: 10.25008/abdiformatika.v1i1.130.

[4] S. Andriyani, M. F. Sidiq, and B. P. Zen, "Analisis Celah Keamanan Pada Website Dengan Menggunakan Metode Penetration Testing Dan Framework Issaf Pada Website SMK Al-Kautsar,"

J. Inform. Inf. Technol., vol. 8798, pp. 1–13, 2023.

[5] E. I. Alwi, H. Herdianti, and F. Umar, "Analisis Keamanan Website Menggunakan Teknik Footprinting dan Vulnerability Scanning," *INFORMAL Informatics J.*, vol. 5, no. 2, p. 43, 2020, doi: 10.19184/isj.v5i2.18941.

[6] A. Hidayah and S. Syahrani, "Internal Quality Assurance System Of Education In Financing Standards and Assessment Standards," *Indones. J. Educ.*, vol. 3, no. 2, pp. 291–300, 2022, doi: 10.54443/injoe.v3i2.35.

[7] J. J. B. H. Yum Thurfah Afifa Rosaliah, "Pengujian Celah Keamanan Website Menggunakan Teknik Penetration Testing dan Metode OWASP TOP 10 pada Website SIM," *Senamika*, vol. 2, no. September, pp. 752–761, 2021.

[8] A. Aryapranata, "Web Application Firewall pada Situs Web Institut Bisnis Nusantara www.ibn.ac.id," *J. Esensi Infokom J. Esensi Sist. Inf. dan Sist. Komput.*, vol. 4, no. 1, pp. 55–59, 2020, doi: 10.55886/infokom.v4i1.321.

[9] D. P. Putranto, J. Jayanta, and B. Hananto, "Analisis Keamanan Website Leads UPNVJ Terhadap Serangan SQL Injection & Sniffing Attack," *Inform. J. Ilmu Komput.*, vol. 18, no. 3, p. 230, 2022, doi: 10.52958/iftk.v18i3.4690.

[10] M. A. Mu'min, A. Fadlil, and I. Riadi, "Analisis Keamanan Sistem Informasi Akademik Menggunakan Open Web Application Security Project Framework," *J. Media Inform. Budidarma*, vol. 6, no. 3, p. 1468, 2022, doi: 10.30865/mib.v6i3.4099.

[11] K. Dhiatama Ayunda *et al.*, "Implementation and analysis ModSecurity on web-based application with OWASP Standards," *J. Tek. Inform. dan Sist. Inf.*, vol. 8, no. 3, pp. 1638–1650, 2021, [Online]. Available: https://jurnal.mdp.ac.id/index.php/jatisi/article/view/1223