PETIR: Jurnal Pengkajian dan Penerapan Teknik Informatika
Vol. 16, No. 1, Maret 2023, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v16i1.1910

Analisis Cyber Crime handling pada Aplikasi Web dengan WAF ModSecurity

Hardianto[1]; Tata Sutabri[1]*

1. Universitas Bina Darma, Sumatera Selatan 30111, Indonesia

[*])Email: hardiantosst@gmail.com

Received: 23 Januari 2023 | Accepted: 14 Maret 2023 | Published: 25 April 2023

ABSTRACT

Utilization of technology, especially the internet in everyday life, plays a very important role. Currently, internet users have reached 5.07 billion people or around 63.45% of the world's total population. The most internet usage is web-based applications so that web applications are the highest target for cyber-attack. The cyber-attack on web applications is increasing both in number and intensity. According to an Acunetix survey about 21% of web applications has very high risk and more than 63% has medium-level vulnerabilities. This paper will discuss the Application of a Web Application Firewall (WAF) on a web server as a solution in handling cyber-crime in web applications. WAF has the ability to filter packets, block HTTP traffic and also logging. The WAF application that will be used is ModSecurity because it is an open source but stable and effective. The test results show that ModSecurity can prevent SQL Injection attack but it does not put too much pressure on Web Server performance.

Keywords: *Internet, Cyber Crime, Web Application Firewall*

ABSTRAK

Pemanfaatan teknologi terutama internet dalam kehidupan sehari-hari mengambil peranan sangat. Saat ini pengguna internet telah mencapai 5,07 miliar orang atau sekitar 63,45% dari total populasi dunia. Penggunaan internet paling banyak adalah aplikasi berbasis web sehingga aplikasi web menjadi sasaran serangan paling tinggi. Serangan terhadap aplikasi web semakin meningkat baik jumlah maupun intensitasnya. Menurut survei Acunetix sekitar 21% aplikasi web mempunyai resiko yang sangat tinggi dan lebih dari 63% mempunyai kerentanan tingkat menengah. Tulisan ini akan membahas Penerapan Web Application Firewall (WAF) pada web server sebagai salah satu solusi dalam penanganan cyber crime pada aplikasi web. WAF memiliki kemampuan untuk memfilter paket, memblokir lalu lintas HTTP dan juga logging. Aplikasi WAF yang akan digunakan adalah ModSecurity karena sifatnya yang open source namun stabil dan efektif. Hasil pengujian menunjukkan ModSecurity dapat menangkal serangan SQL Injection namun tidak terlalu membebani kinerja Web Server.

Kata kunci: *Internet, Cyber Crime, Web Application Firewall*

Vol. 16, No. 1, Maret 2023, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v16i1.1910

1. PENDAHULUAN

Perkembangan teknologi informasi saat ini sangat pesatnya. Pemanfaatan teknologi terutama internet untuk kehidupan sehari-hari semakin mengambil peranan sangat penting. Berdasarkan laporan *We Are Social dan Hootsuite*, jumlah pengguna internet di seluruh dunia pada oktober 2022 telah mencapai 5,07 miliar orang atau sekitar 63,45% dari populasi global yang total mencapai 7,99 miliar orang [1]. Hampir semua bidang pekerjaan yang ada saat ini difasilitasi atau membutuhkan internet. Internet merupakan komunikasi jaringan global yang menghubungkan seluruh komputer di dunia meskipun berbeda sistem operasi dan mesin [2]. Salah satu penggunaan internet adalah penggunaan aplikasi berbasis web. Kemudahan akses ke seluruh penjuru dunia melalui internet adalah salah satu faktor penting dalam penggunaan aplikasi web. Aplikasi berbasis web umumnya terdiri dari 2 komponen penting yaitu *web server* dan halaman web. Berdasarkan survei yang dilakukan oleh Netcraft sampai dengan bulan Desember 2022, aplikasi web server yang paling banyak digunakan adalah Apache [3]. Namun perkembangan teknologi Informasi saat ini seolah-olah menjadi pedang bermata dua, karena selain memberikan kontribusi bagi peningkatan kemajuan, kesejahteraan, dan peradaban manusia namun juga bisa menjadi ancaman bagi keamanan data pribadi [4].

Menurut Data Statistik yang dirilis oleh *Positive Technology* pada tahun 2018, terdapat enam target utama yang menjadi sasaran serangan keamanan *cyber* yaitu institusi di sektor keuangan, perusahaan transportasi, teknologi informasi, hiburan, kesehatan dan institusi pemerintahan [5]. Badan Siber dan Sandi Negara melalui Pusat Operasi Keamanan Siber Nasional mencatat untuk kasus serangan siber paling banyak di Indonesia adalah menyerang aplikasi web terutama untuk situs pemerintah [6].

Serangan terhadap aplikasi web menunjukkan peningkatan setiap tahunnya [7], baik jumlah maupun intensitas serangannya. Keamanan aplikasi web sekarang menjadi perhatian khusus bagi setiap organisasi terlebih aplikasi web tersebut digunakan dalam menjalankan bisnis secara online. Survei Acunetix pada tahun 2021 mencatat bahwa sekitar 21% aplikasi web mempunyai resiko yang sangat tinggi terhadap serangan *Cross-Site Scripting* (XSS) dan *SQL injections*. Bahkan lebih dari 63% aplikasi web mempunyai kerentanan tingkat menengah terhadap serangan *Cross-Site Request Forgery* (CSRF) [8].

Salah satu solusi penanganan terhadap *cyber crime* yang mengancam web aplikasi adalah menggunakan *Web Application Firewall* (WAF) pada *web server* yang digunakan. Penerapan WAF dapat mengurangi resiko keamanan yang terjadi pada aplikasi web dengan cara memfilter paket, memblokir lalu lintas HTTP dan juga logging. Implementasi WAF ini juga diharapkan tidak terlalu berpengaruh terhadap performa dari *web server* dalam melayani *request* dari pengguna. Dalam studi ini akan dibahas implementasi WAF dengan menggunakan modul keamanan aplikasi ModSecurity. ModSecurity adalah salah satu aplikasi berbasis *Open Source* yang paling stabil dan efektif yang dapat dimanfaatkan untuk implementasi WAF [9]. Karena sifatnya yang *Open Source*, ModSecurity dapat menjadi solusi alternatif dengan biaya yang murah dalam memperkuat keamanan aplikasi web. Selain itu akan dibahas juga sejauh mana sebuah WAF mampu diandalkan untuk mengatasi serangan *cyber crime* pada aplikasi web.

2. METODE/PERANCANGAN PENELITIAN

2.1. Dasar Teori

Keamanan Aplikasi Web

Keamanan pada aplikasi web merupakan isu yang menjadi perhatian dan membutuhkan penanganan khusus. Terdapat beberapa pilihan yang dapat digunakan dalam menerapkan layanan

Vol. 16, No. 1, Maret 2023, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v16i1.1910

keamanan aplikasi web walaupun pilihan tersebut tidak sepenuhnya sempurna tetapi menjadi langkah pencegahan dari hal-hal yang tidak dinginkan.

Siklus hidup keamanan aplikasi web dibagi menjadi 3 bagian besar yaitu *Secure Development*, *Secure Deployment*, dan *Secure Operations* [10].

Secure Development

Secure development dimulai dari bagaimana dalam membangun aplikasi web dengan menerapkan prinsip keamanan. Beberapa pendekatan dalam menerapkan *secure development* antara lain dengan penerapan *secure* siklus hidup pengembangan system (SDLC), *static analysis* dan *dynamic analysis*.

Secure Deployment

Tahap berikutnya setelah tahapan pengembangan aplikasi selesai adalah dengan melakukan pengujian dan validasi dari aplikasi yang dihasilkan. Tujuan tahapan ini adalah memastikan bahwa aplikasi tidak memiliki celah keamanan yang serius. Tahap uji dan validasi aplikasi dapat dilakukan dengan metode berikut:

1. *Vulnerability Assessment* dengan melakukan *scanning* pada aplikasi web. Hasil *scanning* dapat mengetahui apakah pada aplikasi web mempunyai celah keamanan.
2. *Penetration Testing*, yaitu proses untuk mengetahui celah keamanan pada aplikasi dengan cara membobol aplikasi untuk menentukan celah keamanan dan resiko yang ditimbulkan.

Secure Operation

Bagian ini adalah bagaimana meningkatkan keamanan aplikasi web dengan cara menerapkan perangkat keamanan seperti *web application firewall* (WAF) atau aplikasi monitoring lainnya pada saat aplikasi web telah dioperasikan.

Web Application Firewall

Web application firewall (WAF) merupakan salah satu metode keamanan pada aplikasi web. WAF menurut *Web Application Security Consortium* (WASC) dapat diartikan sebagai sebuah perangkat perantara yang berada antara *web client* dan *web server* yang berfungsi menganalisis pesan pada OSI *Layer-7* ketika terjadi pelanggaran dalam kebijakan keamanan yang telah ditentukan [11]. Sebuah WAF bisa berbasis jaringan, *host-based* atau *cloud-based*, dan kadang digunakan melalui proxy terbalik di depan sebuah website atau aplikasi.

Lokasi WAF pada topologi jaringan berada di depan atau sebagai pembatas antara jaringan eksternal dan internal. Dalam penggunaan WAF hanya perlu adanya tambahan konfigurasi pada web server tanpa perlu ada perubahan pada *script* aplikasi. Jadi WAF dapat diterapkan pada aplikasi yang sudah beroperasi. Seperti firewall pada umumnya, WAF menyaring masuk dan keluarnya data dan dapat menghentikan atau memblokir lalu lintas jaringan yang dianggap berbahaya sesuai dengan aturan yang telah diterapkan.

WAF menggunakan tiga pendekatan untuk menganalisis dan menyaring konten dari HTTP [12]. Pendekatan tersebut antara lain:

1. *Whitelisting*

   Pendekatan ini akan menolak semua permintaan secara *default* dan hanya mengizinkan permintaan yang sudah dipercaya. Biasanya sudah ada alamat IP yang disediakan dan diketahui aman. Kekurangan pendekatan *whitelisting* adalah adanya kemungkinan WAF akan melakukan blok *traffic* baik secara tidak sengaja. Meskipun bisa sangat efisien dan lebih

Vol. 16, No. 1, Maret 2023, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v16i1.1910

mudah penggunaannya dibandingkan pendekatan *blacklisting*, tapi kadang menggunakan whitelisting menjadi kurang akurat.

2. *Blacklisting*

Pendekatan ini secara *default* akan membiarkan data dan menggunakan preset tertentu untuk memblok *traffic* berbahaya web atau aplikasi web. Sederhananya, *blacklisting* adalah penggunaan peraturan tertentu yang mampu mengindikasikan sebuah bahaya. Pendekatan *Blacklisting* lebih tepat untuk website publik karena banyak menerima *traffic* dari alamat IP yang tidak familiar, dan tidak diketahui apakah itu *traffic* berbahaya atau baik. Kekurangannya adalah dibutuhkannya usaha lebih untuk menggunakannya, dan harus memiliki informasi lebih untuk menyaring data berdasarkan informasi spesifik.

3. *Hybrid Security*

Model ini menggunakan kedua elemen dari Whitelisting dan Blacklisting

Gambar 1. *Lokasi sebuah Web Application Firewal (WAF) dalam Topologi Jaringan*

ModSecurity

ModSecurity sering disebut ModSec merupakan salah satu *Web Application Firewall* (WAF) yang bersifat opensource dan paling banyak digunakan. Modsecurity berfungsi mendeteksi dan mencegah serangan terhadap aplikasi web. Seperti umumnya aplikasi *firewall*, *Modsecurity* berfungsi untuk melakukan filter terhadap data yang masuk dan data yang keluar pada web server.

ModSecurity pertama kali dikembangkan oleh Ivan Ristic pada November 2002 dan dirilis di bawah Lisensi Apache 2.0. Pada awalnya *ModSecurity* hanya dapat digunakan pada server Apache saja namun pengembangan berikutnya dapat digunakan juga pada server Nginx dan IIS.

ModSecurity bekerja dengan mendeteksi dan memblokir permintaan yang dianggap berbahaya, berdasarkan rule konfigurasi yang disebut SecRules. Firewall ini menyediakan dua opsi pengembangan [13]. Opsi pertama adalah penerapan sebagai *Reverse proxy*. Penerapan sebagai *reverse proxy* akan menyediakan *ModSecurity* sebagai proxy server yang terpisah yang berdiri di antara web server dan clientnya. Keuntungan dari pengaturan opsi ini adalah melindungi lebih banyak web server pada saat yang sama dan mengisolasi dari system. Opsi kedua adalah *embedded deployment* yang lebih sederhana karena tidak perlu untuk mengubah arsitektur system yang dilindungi. Beberapa fungsi dari ModSecurity antara lain [14]:

a. Monitoring keamanan dan akses kontrol
b. *Virtual patching*
c. Perekaman seluruh trafik HTTP
d. *Security assessment*
e. *Web application hardening*
f. *Passive security assessment*
g. *Simple request or Regular expression based Filtering*
h. *URL Encoding Validation*
i. *Auditing*

Vol. 16, No. 1, Maret 2023, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v16i1.1910

j. *IP Reputation*
k. *Null byte attack prevention*
l. *Server identity masking*
m. *Uploads memory limits*
n. *Protect web from Brute-force attacks & SQL injection*

ModSecurity mencatat semua informasi terkait serangan dalam sebuah file log *modsec_audit.log* dan Apache log dalam file *error.log*. Dalam operasionalnya rekaman pencatatan tersebut dapat diekspor ke dalam format IDMEF atau *Intrusion Detection Message Exchange Format* yaitu format standar untuk peringatan atau deskripsi serangan. Format ini digunakan dalam banyak system yang berfungsi untuk mendeteksi kegiatan yang tidak biasa. Format IDMEF adalah representasi beriorentasi objek yang menggunakan Bahasa XML yang memudahkan manipulasi elemen di dalamnya.

2.2. Metode Penelitian

Metode penelitian yang digunakan dalam penelitian ini adalah metode penelitian experimental. Metode penelitian eksperimental merupakan bagian dari metode penelitian kuantitaif dimana penelitian dilakukan dengan serangkaian ujicoba untuk menemukan efek dari perlakuan tertentu terhadap obyek penelitian. Selain itu juga digunakan penelitian kepustakaan (*literature review*). Metode ini digunakan untuk menggali sumber sumber data yang dihasilkan dari beberapa riset, laporan maupun literatur, antara lain jurnal ilmiah, artikel, dan tulisan lain yang berhubungan dengan judul penelitian.

Makalah ini menyajikan hasil pengujian ModSecurity terhadap 10 risiko teratas OWASP untuk mengevaluasi keefektifannya dalam mendeteksi serangan dan performa pada web server.

3. HASIL DAN PEMBAHASAN

Dalam pengujian penelitian ini untuk melihat efektivitas dari *ModSecurity* dalam melakukan pencegahan serangan terhadap aplikasi web digunakan sebuah *website* yang dijadikan target serangan. Aplikasi web tersebut adalah *mutillidae*. *Mutillidae* adalah aplikasi web yang bersifat *free* dan *open source* yang sengaja dirancang dengan memiliki banyak celah keamanan sehingga dapat dijadikan target pengujian serangan.

Tahapan pengujian dalam penelitian ini terdiri atas proses persiapan, desain dan implementasi, testing dan analisis untuk menggambarkan kesimpulan.

Proses Persiapan

Dalam proses persiapan ini dilakukan studi literatur dan pencarian informasi terhadap hal-hal yang akan dilakukan dalam penelitian. Topik literatur yang dipilih ada topik yang membahas terkait dengan *Web Application Firewall* (WAF), pentingnya WAF untuk meningkatkan fungsi keamanan pada aplikasi web dan cara implementasinya pada aplikasi web.

Desain dan Implementasi

Tahapan ini membuat perancangan lingkungan penelitian dengan membuat topologi jaringan yang diikuti dengan pengembangan desain lingkungan. Kemudian dilakukan implementasi penelitian.

1. Perancangan Topologi Jaringan

PETIR: Jurnal Pengkajian dan Penerapan Teknik Informatika
Vol. 16, No. 1, Maret 2023, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v16i1.1910

Perancangan topologi jaringan merupakan dasar dari pengembangan lingkungan penelitian. Dalam topologi ini terdiri atas 3 obyek yaitu *attacker* atau penyerang, perangkat WAF *ModSecurity* dan web server.

2. Pembangunan lingkungan penelitian

Untuk implementasi dan pengujian dalam penelitian ini digunakan sebuah PC sebagai web server dan sebuah laptop yang digunakan untuk melakukan penyerangan. Spesifikasi perangkat yang digunakan adalah sebagai berikut:

i. Mesin target yang akan berfungsi sebagai web server akan berjalan pada system operasi windows Server 2016, RAM 4 GB, Xampp sebagai web server Apache.
ii. Mesin yang digunakan untuk penyerangan adalah mesin *host* atau laptop dengan system operasi Windows 10 64 bit

Langkah selanjutnya adalah dilakukan instalasi berikut:

1) Konfigurasi XAMPP sebagai Apache *Web Server*
2) Instalasi *ModSecurity* sebagai modul WAF pada *web server*
3) Membuat aturan konfigurasi pada *ModSecurity* menggunakan *Open Web Application Security Project Core Rule Set* (OWASP CRS).

Testing

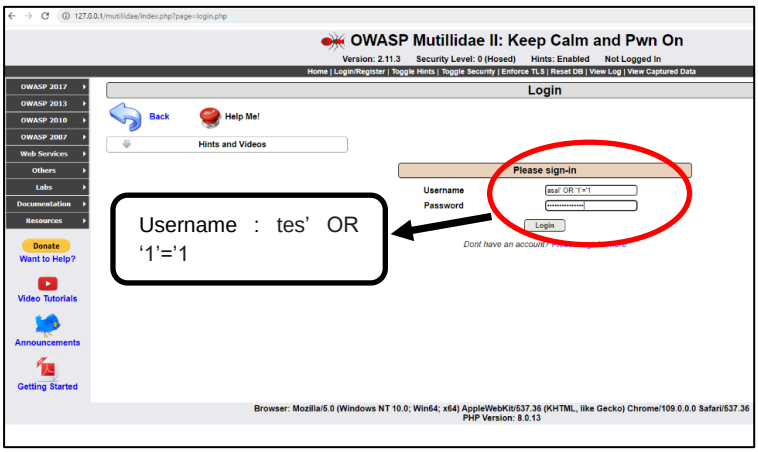
Dalam penelitian lain disebutkan bahwa implementasi WAF dalam aplikasi web akan meningkatkan fungsi keamanan sehingga mencegah kerusakan dari serangan yang terjadi pada aplikasi web. Pengujian dalam penelitian ini dilakukan dalam dua kondisi. Pertama adalah serangan pada aplikasi web dengan tidak menerapkan *ModSecurity* dan kondisi kedua adalah serangan pada aplikasi web yang menerapkan *ModSecurity*.

i. Pengujian terhadap serangan *SQL Injection*

Pada pengujian ini dilakukan serangan *SQL Injection* pada aplikasi web. Beberapa input *SQL Injection* yang digunakan untuk serangan:

- tes' OR '1'='1, serangan untuk melakukan *bypass* terhadap form login.
- 'union select 1,2,@@version,4 #, uji *SQL Injection* untuk mengetahui versi *MySQL* yang digunakan
- 'union select 1,2,@@datadir,4 #, pengujian *SQL Injection* untuk mengetahui direktori database.

Pada gambar 2 memperlihatkan serangan *SQL Injection* untuk *bypass* terhadap form login.



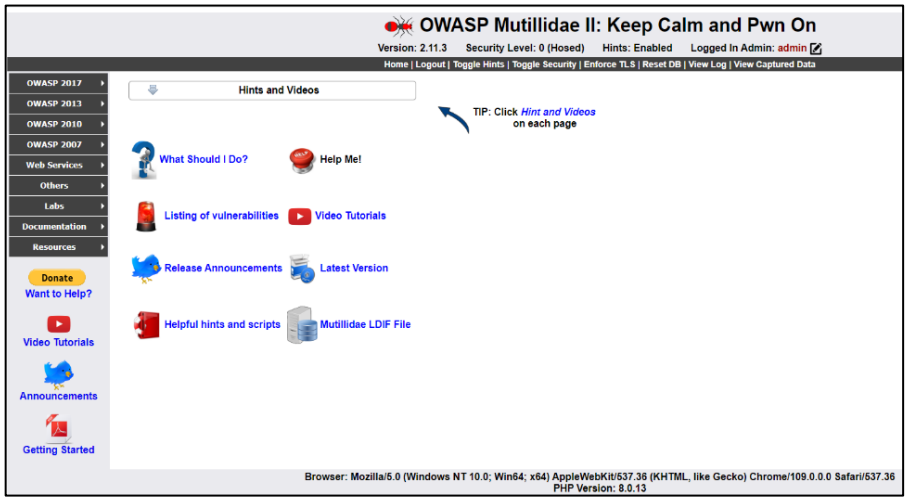
Gambar 2. *Input SQL Injection pada aplikasi web*

Vol. 16, No. 1, Maret 2023, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v16i1.1910

Hasil Pengujian:

1) *ModSecurity* tidak diterapkan

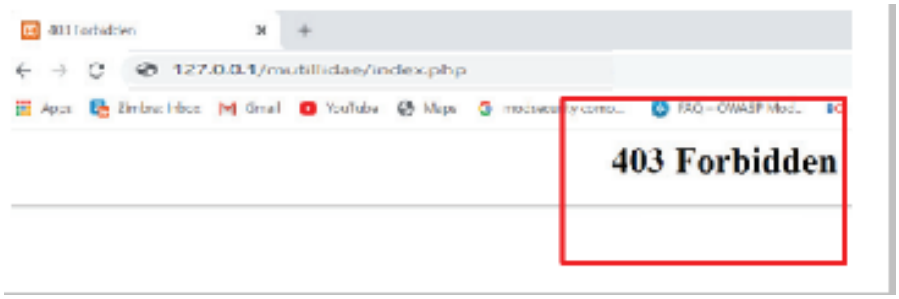
Serangan *SQL Injection* pada aplikasi web yang belum menerapkan *ModSecurity* berhasil dan penyerang dapat melakukan login menggunakan akun dengan level admin. Hal ini terlihat pada gambar 3 dapat masuk ke dalam halaman aplikasi web.



Gambar 3. *Serangan SQL Injection berhasil login dengan akun level admin*

2) *ModSecurity* diterapkan

Hasil dari serangan SQL Injection di aplikasi web dengan *ModSecurity* diterapkan tidak berhasil, ditunjukkan oleh aplikasi web menanggapi dengan kode 403 / Terlarang, yang mencegah akses diberikan pada upaya login, seperti yang ditunjukkan pada Gambar 4. *ModSecurity* akan melakukan perlindungan terhadap perintah *SQL Injection* yang telah kita tetapkan dengan mengenali suatu *request* berdasarkan *regular expression* yang ada di *core rule*. Apabila *request* terdapat perintah *SQL Injection* yang *match* dengan aturan filter yang ada di *core rule* maka *request* tersebut akan ditolak atau tidak diteruskan.



Gambar 4. *Input SQL Injection pada aplikasi web setelah diterapkan ModSecurity*

ii. Pengujian *Web Load*

Pengujian ini melihat apakah ada perbedaan yang signifikan terhadap kecepatan web server dalam memberikan respon semua *request* yang masuk ke aplikasi web sebelum dan setelah diterapkan *ModSecurity*. Hasil pengujian menunjukan rata-rata *response time web server* setelah diterapkan *ModSecurity* masih relatif sama yaitu 30-50 ms namun meningkat ketika ada *request*

Vol. 16, No. 1, Maret 2023, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v16i1.1910

mnjadi 80 ms. Hal ini menunjukkan bahwa setelah adanya penerapan *ModSecurity* terjadi pengurangan kecepatan *web server* dalam menerima *request* namun tidak terlalu signifikan.

4. KESIMPULAN DAN SARAN

Salah satu solusi yang dapat digunakan untuk mencegah ancaman terhadap aplikasi web adalah dengan mengimplementasikan *Web Application Firewall* (WAF). WAF memiliki kemampuan untuk memfilter paket, memblokir lalu lintas HTTP dan juga logging. *ModSecurity* adalah salah satu modul keamanan aplikasi web berbasis Open Source yang paling stabil dan efektif yang dapat dimanfaatkan untuk implementasi WAF. Ada aturan di ModSecurity yang bisa dikonfigurasi untuk mencegah ancaman yang terjadi di web aplikasi. Karena sifatnya Open Source, ModSecurity dapat menjadi solusi alternatif dengan biaya yang murah memperkuat keamanan aplikasi web. Berdasarkan hasil pengujian terhadap penerapan *ModSecurity* dapat disimpulkan:

1. Instalasi dan Setting *ModSecurity* pada web server mudah dilakukan
2. *ModSecurity* tidak membebani kinerja *web server* secara berlebihan
3. Penerapan *ModSecurity* dengan menggunakan *core rule* OWASP terbukti dapat mecegah serangan *SQL Injection* pada aplikasi web.

DAFTAR PUSTAKA

[1] "Jumlah Pengguna Internet Global Tembus 5 Miliar Orang pada Oktober 2022," Katadata Media Network, [Online]. Available: https://databoks.katadata.co.id/datapublish/2022/. [Diakses 14 January 2022].

[2] T. S. P. N. R. N. d. R. P. Tata Sutabri, "Pelatihan Pemanfaatan Internet Sehat Bagi Masyarakat Desa Cibinuang Kabupaten Kuningan," Jurnal Pengabdian Masyarakat, vol. 3 No. 2, pp. 847-853, 2022.

[3] "Web Server Survei," Netcraft, 20 Desember 2022. [Online]. Available: https://news.netcraft.com/archives/category/web-server-survey/. [Diakses 14 Jan 2023].

[4] T. Sutabri, Pengantar Teknologi Informasi, 1 ed. Andi, 2014.

[5] "Attacks on web applications: 2018 in review," Positive Technology, 26 Juni 2019. [Online]. Available: https://www.ptsecurity.com/wwen/analytics/web-application-attacks-2019/. [Diakses 14 Januari 2023].

[6] "Rekap Serangan Siber (Januari – April 2020), Badan Siber dan Sandi Negara," Security Advisory, 20 April 2020. [Online]. Available: https://bssn.go.id/rekap-serangan-siber-januari-april-2020/. [Diakses 15 Januari 2023].

[7] "Statistik Insiden Keamanan Internet Indonesia," GOV CSIRT, November 2012. [Online]. Available: http://govcsirt.kominfo.go.id. [Diakses 15 Januari 2023].

[8] "Web Application Vulnerability Report 2021," Acunetix Team, 2021. [Online]. Available: https://www.acunetix.com/white-papers/acunetix-web-application-vulnerability-report-2021/?#executive-summary. [Diakses 15 Januari 2023].

[9] M. L. a. D.-S. P. S. Prandl, "A Study of Web Application Firewall Solutions," ICISS, 2015.

[10] &. l. A. Mogul R., Building a Web Application Security Program, Pheonix AZ: Securosis, L.L.C, 2009.

[11] "Web Security Glossary," Web Application Security Consortium, 23 Juni 2013. [Online]. Available: http://www.webappsec.org/projects/glossary/v1/wasc_glossary_02262004.pdf. [Diakses 14 Januari 2023].

Vol. 16, No. 1, Maret 2023, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v16i1.1910

[12] "Apa itu Web Application Firewall (WAF) ?," Indonesian Cloud, [Online]. Available: https://indonesiancloud.com/apa-itu-web-application-firewall-waf/. [Diakses 15 Januari 2023].

[13] "Modsecurity : Open source web application firewall," T. SpiderLabs, [Online]. Available: http://modsecurity.org. [Diakses 15 Januari 2023].

[14] "Apa itu ModSecurity," Masterweb.com, 22 Mei 2020. [Online]. Available: https://blogs.masterweb.com/apa-itu-modsecurity/. [Diakses 15 Januari 2023].