ANALISA KEAMANAN INFORMASI PADA APLIKASI BERBASIS WEB MENGGUNAKAN TEKNIK WEB APPLICATION FIREWALL MODSECURITY

Albi Alamsyah [1], Triawan Adi Cahyanto, M.Kom [2]
1. Mahasiswa 2. Dosen Pembimbing
Teknik Informatika, Universitas Muhammadiyah Jember
Tegal Banteng, Wuluhan Jember
082333200303, albialamsyah11@gmail.com

Abstrak

Perkembangan internet mengubah dampak bagi kehidupan masyarakat segala seuatu informasi cepat terselesaikan. Danpak adanya perkermbangan internet berdampak pula pada perkembangan dunia penyerangan/ *hecker* pada sistem informasi/ *website* yang merupakan hasil dari perkembangan internet, terdapat banyak website menjadi korban penyerangan oleh *hecker*/peretes. Oleh karena itu soal keamanan informasi menjadi topik dalam dunia teknologi informasi Terkait celah keamaman pada aplikasi *web*, biasanya celah tersebut muncul karena adanya *bug* atau kesalahan pemrograman yang lupa untuk diatasi oleh pemrogram. Misalkan lupa untuk melakukan *filter* suatu masukan, sehingga jika memasukkan karakter-karakter berbahaya seperti tanda petik(‘) akan mengakibatkan kesalahan dan munculnya sebuah *error* pada halaman *web* dan biasanya, hal tersebut dimanfaatkan oleh peretas untuk mencoba mendapatkan informasi dari *error* tadi. Oleh karena itu dilakukan pendeteksian dan pengamanan serangan *web,* pada penelitian ini akan dibangun sistem untuk mendeteksi serengan-serangan terhadap *web* serangan-serangan tersebut adalah *Cross-Site Scripting (XSS), SQL Injection* dan *Remote File Inclusion (RFI)* Metode pendeteksian terhadap serangan tersebut menggunakan *Web Application Firewall* dan *Modsecurity.* Walaupun nilai dari sebuah keamanan tidak ada yang mutlak, harapannya dengan sistem yang akan dibangun nantinya dapat mengurangi peluang seorang peretas untuk dapat meretas aplikasi *web.*
Kata Kunci : *hecker, -Site Scripting (XSS), SQL Injection* dan *Remote File Inclusion (RFI), Web Application Firewall* dan *Modsecurity*

I. PENDAHULUAN

Pada era saat ini, *internet* sudah menjadi suatu kebutuhan yang harus terpenuhi untuk mencari atau memperoleh informasi. Segala kemudahan bisa tercapai dengan adanya *internet* pada saat ini. Berdasarkan data dari Kementerian Komunikasi dan Informatika (Kemenkominfo) pengguna *internet* diindonesia pada tahun 2015 adalah 150 juta orang, atau sekitar 61% dari populasi indonesia, ditambah dengan perkembangnya aplikasi *web* yang terhubung dengan basis data seperti *toko online*, *social networks, website sistem informasi penting* dan sebagainya. Hal ini berakibat kuaitas dan implikasi dari serangan dari internet yang beberapa tahun terakhir menjadi target serangan *hacker.* Berdasarkan data dari *The Open Web Application Security Project* (OWASP) pada tahun 2013, yang melakukan *survey* mengenai ancaman yang sering terjadi pada aplikasi web diantaranya merupakan ancaman *Cross-Site Scripting (XSS), SQL Injection, Local File Inclusion(LFI),* dan *Remote File Inclusion (RFI).*

Oleh karena itu dengan berkembangnya teknologi *web*, faktor keamanan informasi tentunya menjadi suatu perhatian yang penting seperti dengan memasang *firewall.*Terkait celah

keamaman pada aplikasi *web*, biasanya celah tersebut muncul karena adanya *bug* atau kesalahan pemrograman yang lupa untuk diatasi oleh pemrogram. Misalkan lupa untuk mem-*filter* suatu masukan, sehingga jika memasukkan karakter-karakter berbahaya seperti tanda petik(‘) akan mengakibatkan kesalahan dan munculnya sebuah *error* pada halaman *web* dan biasanya hal tersebut dimanfaatkan oleh peretas untuk mencoba mendapatkan informasi dari *error.* Untuk itu diperlukan pengamanan dan pendeteksian dari segala bentuk usaha percobaan serangan terhadap aplikasi berbasis *web* dengan cara melakukan pencocokan terhadap *rule* atau polapola serangan. *Rule* atau pola-pola serangan tersebut dicocokkan dengan data *request HTTP.*

Berdasarkan kasus tersebut maka akan dilakukan pengamanan dan pendeteksian pada aplikasi *web* dengan Metode pengamanan yang akan di terapkan yaitu menggunakan *Web Application Firewall* dan *Modsecurity*. *Web Application Firewall*(WAF) memiliki beberapa fungsi, mulai dari monitoring trafik, *secure directory*, pemfilteran *string* dan proteksi terhadap serangan seperti *SQL Injections*, *Cross-Site Scripting*, dan *Remote File Inclusion*. Sedangan *Modsecurity* seperti *firewall* pada umumnya memiliki tugas untuk melakukan pemfilteran pada data yang masuk maupun keluar, dan melakukan blocking traffic yang dianggap berbahaya sesuai dengan rule yang ditetapkan. Setelah itu segala bentuk serangan yang telah terdeteksi akan disimpan pada suatu database.

Aplikasi *web* yang akan diuji yaitu sampel data dari *website* pentesterlab.com nantinya akan diuji tingkat keamanan terhadap serangan seperti *SQL Injections*, *Cross-Site Scripting*, dan *Remote File Inclusion* pada apliasi *web* tersebut, dengan menggunakan metode pengamanan dan pendeteksian serangan dengan metode *Web Application Firewall* dan *Modsecurity*. Dengan adanya kedua pengamanan tersebut makan dapat dilakukan analisa tingkat dan keakuratan keamanan aplikasi *web*d engan menggunakan dua metode tesebut khususnya serangan-serangan *SQL Injections*, *Cross-Site Scripting* dan *Remote File Inclusion.*

Berkaitan dengan hal tersebut, melalui penelitian ini Hasil yang diharapkan nantinya adalah serangan-serangan seperti *SQLInjection, Cross Site Scripting (XSS), Local File Inclusion(LFI)* dan *Remote File Inclusion (RFI)* yang dapat membahayakan kerahasiaan, integritas dan ketersediaan yang diberikan oleh *web* tersebut dapat sedikit lebih aman.

II. METODE PENELITIAN

Tahapan yang dilakukan dalam penelitian ini adalah

a. Implementasi *WAF* Modesecurity
b. Pengujian Serangan

III. IMPLEMENTASI DAN PENGUJIAN

Pada tahap ini akan dilakukan perancangan *web application firewall* modsecurity. WAF ini akan tertanam dan dijalankan pada sistem operasi linux ubuntu yang mana nantinya sebagai

deteksi serangan *SQL Injections*, *Cross-Site Scripting* dan *Remote File Inclusion.*

A. Uji Coba

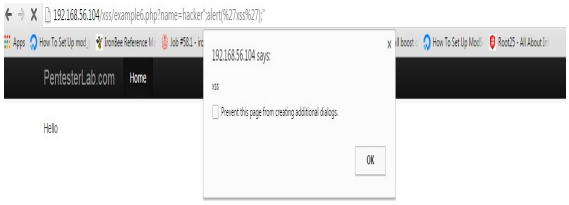
Pengujian *SQL Injections*

Gambar 3 Serangan SQL Injection 1 tanpa *WAF*

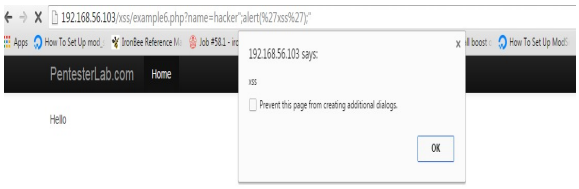
Gambar 4 Serangan SQL Injection 1 dengan *WAF*

Pengujian *Cross-Site Scripting*

Pada uji coba ini WAF Modsecurity ini setelah dilakukan 5 kali percobaan di *web pentesterlab* tidak bisa menangani serangan injeksi *cross-site scripting* atau XSS atau mengbloking serangan injeksi *cross-site scripting* atau XSS



Gambar 5 Serangan XSS tanpa *WAF*

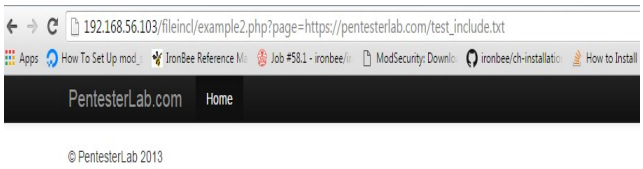


Gambar 6 Serangan XSS dengan *WAF*

Pada uji coba serangan XSS ini modsecurtiy tidak bisa menangani dan mengblok serangan XSS seperti pada gambar di atas IP 192.168.56.104 merupakan web pentesterlab tanpa WAF modsecurity sedangan IP 192.168.56.103 menggunakan WAF modsecurity

Pengujian *Remote File Inclusion*

Gambar 7 serangan *Remote File Inclusion* tanpa *WAF*



Gambar 8 serangan *Remote File Inclusion* dengan *WAF*

IV KESIMPULAN DAN SARAN

A. KESIMPULAN

Pengamatan telah dilakukan dari hasil uji coba sistem, dapat ditarik kesimpulan sebagai berikut:

1. Pada implementasi ini terdapat 2 buah server *web pentesterlab* yang berhasil dibangun dimana terdapat 1 web yang ternam *Web Application Firewall modsecurity* yaitu dengan IP IP 192.168.56.104 merupakan web

pentesterlab tanpa WAF modsecurity sedangan IP 192.168.56.103 dengan tertanam *Web Application Firewall modsecurity.*

2. 2 buah *web pentesterlab* dilakukan 2 uji coba serangan SQL Injections, XSS dan *Remote File Inclusion.*
3. Pada uji coba ini *Web Application Firewall modsecurity* tidak berhasil memblok /mendeteksi serangan *Cross-Site Scripting/XSS* .

B. SARAN

Berikut merupakan beberapa saran untuk pengembangan sistem di masa yang akan datang, berdasar pada hasil perancangan, implementasi, dan pengujian yang telah dilakukan:

1. Penelitian selanjutnya diharapkan tidak hanya menguji aplikasi website yang berbasis *PHP* dan basis data *MySQL* tetapi dapat menguji aplikasi website yang dibangun bahasa lain seperti *Java, Python*, dan lain sebagainya dan menggunakan basis data selain *MySQ*L.
2. Penelitian selanjutnya dapat dilakukan dengan penambahan teknik penyerangan yang lain, karena perkembangan dunia heaking sangat berkembang masih terdapat banyak teknik penyerangan yang berkembang pada zaman ini.

DAFTAR PUSTAKA

[1] Pribadi, Harijanto.2008. *Firewall melindungi jaringan dari DdoS menggunakan LINUX+MIKROTIK*. Penerbit Andi : Yogyakarta.

[2] Ahmad Muammar. W. K.2004. *FireWall.* Ilmukomputer.com

[3] Sarno, R. & Iffano, I. 2009. *Keamanan Sistem Informasi*. ITS press.

[4] Ellysa. R, Husni. Muchammad, Baskoro Adi Pratomo(2013). Pendeteksi Serangan SQL Injection Menggunakan Algoritma SQL Injection Free Secure pada Aplikasi Web, Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

[5] Rahmat. Fajri, Mazharuddin S. Ary, Studiawan. H (2013). sistem Pendeteksi dan Pencegah Peretasan Terhadap Aplikasi Berbasis *Web* dengan Teknik *Web Application Firewall* (*WAF*), Institut Teknologi Sepuluh Nopember (ITS) Surabaya. ISSN: 2337-3539 (2301-9271 Print).

[6] Garfinkel, S. & Howard, S.E. 2001. Web Security & Commerce. United States: O'Reilly & Associates.

[7] *Chan, Y.B., Yoke, C.A., & Yousefi, D. 2013.* An Exploratory of Airline E-ticker Purchasing Intenttion among Foreign Undergraduates in Malaysia. *Journal of Human and Social Science Research Vol. 1, No. 1 (2013), 51-61.*

[8] Stuttard, D., Pinto, M. 2011. *The Web Application Hacker's Handbook: Finding And*
Exploiting Security Flaws. 2 nd Edition. Canada: John Wiley & Sons, Inc.

[9] Herrmann, S.D. 2002. *A Pratical Guide to Security Engineering And Information Assurance*. New York: Auerbach Publications.

[10] Clarke, J. 2009. *SQL Injection Attacks dan Defense*. Burlington : Syngress Publishing, Inc.

[11] Grossman, J., Hansen, R., Petkov, D.P., & Rager, A. 2007. *XSS Attacks: Cross Site Scripting Exploits and Defense*. Burlington: Syngress Publishing, Inc.

[12] Krawczyk, P. 2013. Most common attacks on web applications. http://ipsec.pl/webapplication-security/most-common-attacks-web-applications.html. (diakses 02 April 2016).

[13] Ande. 2011. Local File Inclusion (LFI). https://evilzone.org/tutorials/localfile-inclusion-(lfi). (02 April 2016).

[14] Christopher Alfeld et al. Ironbee Open Source WAF. https: //www. ironbee. com/ docs/ manual/ ironbee- reference manual. Html. (10 april 2016)

[15] ISO/IEC (2005). Information technology — Security techniques — Code of practice for information security management. http://www.specon.ru/files/ISO IEC%2017979%20(second%20edition).pdf. (10 april 2016)