IMPLEMENTASI KEAMANAN APLIKASI WEB

DENGAN WEB APPLICATION FIREWALL

Risma Yanti Jamain[1] Periyadi S.T,M.T.[2] Setia Juli Irzal Ismail, S.T., M.T.[3]

[1,2,3]Prodi D3 Teknik Komputer, Fakultas Ilmu Terapan, Universitas Telkom

[1]rismayanti.jamain@gmail.com, [2] periyadi@tass.telkomuniversity.ac.id, [3]jul@tass.telkomuniversity.ac.id

ABSTRAK

Aplikasi web sekarang ini semakin berkembang secara pesat. Dengan internet, segala informasi yang ada pada aplikasi web menjadikan aplikasi web itu menjadi incaran para *hacker*. Hal tersebut, membuat web rentan dan membutuhkan sistem keamanan yang dapat melindungi informasi dari web.

Karena adanya kebutuhan akan keamanan applikasi web maka penulis ingin membuat suatu sistem yang dapat mendeteksi dan mencegah serangan dari para *hacker*. Pada proyek akhir ini, serangan yang akan dibahas adalah, *SQL Injection, Cross-Site Scripting, Command Execution.*

Dengan sistem yang dibangun, dapat mengurangi peluang para *hacker* untuk melakukan peretasan terhadap aplikasi web. Untuk membangun sistem, akan digunakan *nginx* sebagai *web server* dan *naxsi* sebagai *web application firewall,* yang memiliki tugas melakukan pemfilteran terhadap apa yang melaluinya dan melakukan *blocking* terhadap apa yang dianggap berbahaya sesuai dengan *rule* yang ditetapkan.

Kata Kunci: Keamanan Aplikasi Web, *Web Application Firewall, Naxsi*

ABSTRACT

Web application nowadays develop fast. Using internet, every information within any web application can be hacker targets. Those reason, making web vulnerable and needing a security system to protect the information from the web.

Because of the needs of web application so the writer is gonna make a system that can detect and prevent an attack from the hacker. In this last project, the attack that will be disscused are sql injection, cross site scripting, and command execution.

Using the system which built, can minimaze the risk of a hacker attack upon the web application which used nginx as web server and naxsi as web application firewall, whose duties are to filter everything that goes through it and block every threatening condition according to the rule that was made.

Keywords: Security of Web Application, Web Application Firewall, Naxsi

1. PENDAHULUAN

Layanan Internet memungkinkan pengguna berbagi layanan bersama dan saling terkait melalu aplikasi web yang ada. Segala informasi dapat dengan mudah didapatkan dari aplikasi web. Adanya teknologi informasi saat ini menjadikan aplikasi web sebagai incaran para *hacker.* Beberapa ancaman yang sering terjadi pada aplikasi web diantaranya *SQL Injection, Cross-Site Scripting* dan *Command Execution.*

SQL (Structure Query Language) Injection adalah teknik yang memanfaatkan kode yang terdapat di dalam program sebuah situs yang lemah tanpa perlindungan yang kuat dari sebuah admin situs tersebut. *Cross-Site Scripting* atau sering dikenal dengan XSS adalah ancaman yang mengizinkan kode (*client side script*) dimasukan ke dalam suatu *website* yang dapat dijalankan pada sisi *user. Command Execution* adalah *bug* yang memungkinkan *attacker* untuk menjalankan perintah-perintah secara *remote* melalui *url.*

2. DASAR TEORI

2.2 SQL Injection

SQL Injection adalah metode ancaman yang mengizinkan *client* untuk mengeksekusi database melalui URL dan mendapatkan akses untuk memperoleh informasi penting. Mekanisme ancaman dilakukan dengan memanfaatkan kesalahan pada kode program yang tidak di*filter*, sehingga menyebabkan terjadinya eksploitasi pada database [**2**]

2.3 Cross-Site Scripting

Cross Site Scripting adalah metode ancaman yang memaksa situs web untuk menampilkan kode berbahaya, yang dijalankan pada *browser* web pengguna. Mekanisme ancaman dilakukan dengan memanfaatkan kesalahan pada kode program yang tidak di-*filter*. Kode tersebut akan dimasukkan ke dalam *webform*, di antaranya *form* buku tamu atau *form* pencarian [**2**].

2.4 Command Execution

Injeksi perintah serangan di mana tujuannya adalah pelaksanaan perintah sewenang-wenang pada sistem operasi host melalui aplikasi rentan. Serangan injeksi perintah yang mungkin ketika aplikasi melewati disediakan pengguna data yang tidak aman (bentuk, cookies, HTTP header dll) untuk shell sistem. Dalam

serangan ini, perintah sistem operasi penyerang yang disediakan biasanya dijalankan dengan hak istimewa dari aplikasi rentan. Serangan injeksi perintah yang mungkin sebagian besar karena validasi input tidak cukup.

Serangan ini berbeda dari Kode Injection, dalam kode injeksi memungkinkan penyerang untuk menambahkan kode sendiri yang kemudian dieksekusi oleh aplikasi. Dalam Kode Injection, penyerang memperluas fungsi default aplikasi tanpa perlu mengeksekusi perintah sistem[**3**].

2.5 Web Server Nginx

Gambar 1.1 Nginx

Nginx dengan cepat memberikan konten statis dengan penggunaan efisien sumber daya sistem. Hal ini dapat menyebarkan dinamis *HTTP* konten di jaringan menggunakan *FastCGI handler* untuk *script*, dan dapat berfungsi sebagai perangkat lunak yang sangat mampu penyeimbang beban. Nginx menggunakan *asynchronous-event* pendekatan untuk menangani permintaan yang diprediksi memberikan kinerja yang lebih bawah beban, kontras dengan *Apache HTTP server model* yang menggunakan berulir atau proses yang berorientasi pada pendekatan-permintaan penanganan [**7**].

2.6 Web Application Firewall

Web Application Firewall (WAF) adalah Suatu metode untuk pengamanan pada aplikasi web, yang berupaya mencegah adanya ancaman dari *attacker* [**8**].

Web Application Firewall dapat bekerja dengan terlebih dahulu melakukan konfigurasi tambahan pada web *server* dan tidak perlu melakukan perubahan pada *script* pembangun aplikasi, sehingga dapat diterapkan pada aplikasi yang sudah berjalan. Seperti *firewall* pada umumnya yaitu melakukan *filter* data masuk dan keluar dan dapat untuk menghentikan *traffic* yang dianggap berbahaya sesuai dengan *rule* yang ditetapkan.

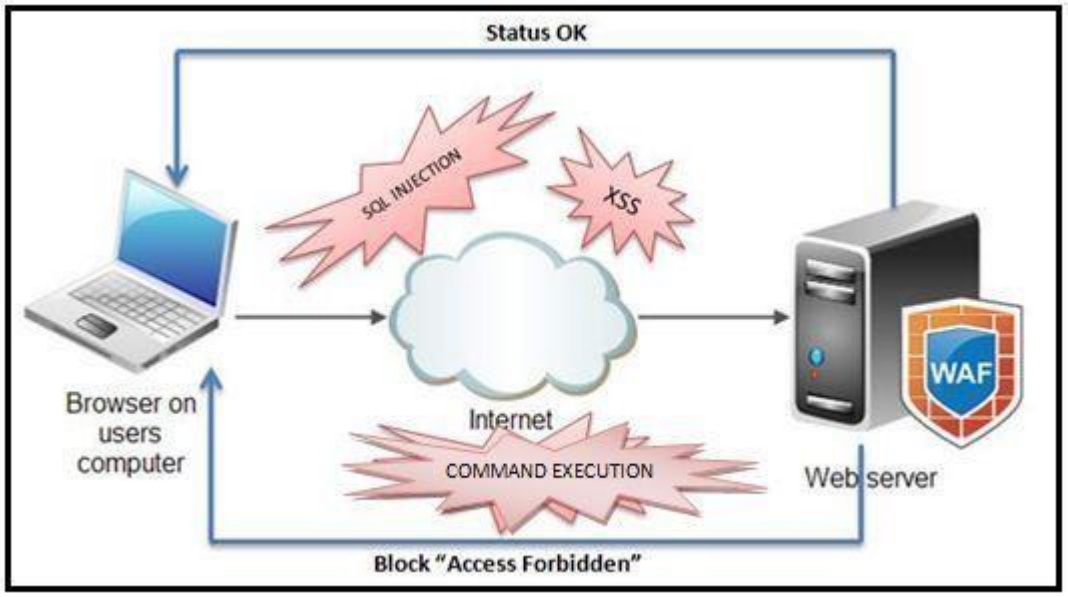
Web Application Firewall memiliki beberapa fungsi, mulai dari monitoring trafik, *secure directory*, pemfilteran *string* dan proteksi terhadap serangan seperti *SQL Injections*, *Cross-Site Scripting*, dan *Unrestricted File Upload*. *Web Application Firewall* membentuk lapisan keamanan yang dapat mendeteksi dan mencegah serangan pada aplikasi web. Adapun tindakan yang dapat dilakukan seperti menghentikan

request dengan status 403 *forbidden* dan juga dapat melakukan *virtual patching.* Dimana *virtual patching* merupakan suatu *rule* yang diterapkan untuk melakukan *patch* tanpa menyentuh aplikasi guna memblokir *request* yang berbahaya.

2.7 Damn Vulnerable Web Application

DVWA adalah aplikasi yang ditujukan untuk para Security Profesional untuk melakukan test terhadap skill yang mereka miliki, DVWA juga cocok bagi anda yang ingin mempelajari beberapa tehnik web-hacking terhadap aplikasi PHP/MySQL, seperti SQL injection, Remote Command Execution, dan lain-lain[**10**].

3. SIMULASI



Gambar 3. 1 Arsitektur Web dengan Implementasi *Naxsi*

Sistem ini akan dibangun dengan menggunakan *web application firewall naxsi*, sebuah web aplikasi firewall opensource yang diimplementasikan pada Linux Mint, dimana pada *naxsi* dikonfigurasikan *basic rule* sebagai parameter pengidentifikasi serangan. Serangan dilakukan pada DVWA, awalnya serangan dilakukan tanpa WAF, *attacker* dapat mengambil informasi pada Web. Pada saat Naxsi diaktifkan, *attacker* melakukan serangan, sedangkan *naxsi* melakukan penyaringan terhadap setiap request HTTP yang akan menuju server. Sehingga bila pada request HTTP mengandung serangan maka request tersebut akan di-*block* atau dihentikan.

4. KESIMPULAN

Ancaman *SQL Injection, Cross-Site Scripting* dan *Command Execution* dapat dicegah dengan menggunakan *web application firewall. Naxsi* dapat digunakan sebagai *web application firewall* yang dapat menghentikan serangan tersebut

DAFTAR PUSTAKA

[1] A. Zaki and S. D. Community, *Kiat Jitu Membuat Web Site tanpa Modal*. Jakarta: PT Elex Media Komputindo, 2009.

[2] Digdo, G. P. *Analisis Serangan dan Keamanan pada Aplikasi Web*. Jakarta: Elex ,edia Komputindo, 2012.

[3] Ariyus,Doni M.Kom, *Sistem Penyusupan pada Jaringan Komputer*. Yogjakarta: Andi, 2007.

[4] A. Sularso, N. Hendrarini, and S. N. M. P. Simamora, *Network Security*. Bandung: Politeknik Telkom, 2009.

[5] Anonim, *Pemrograman Web*. Bandung: Politeknik Telkom, 2009.

[6] A. Rudiyanto, *Pemrograman Dinamis menggunakan PHP dan MySQL*. Yogyakarta: Andi, 2009.

[7] J. Karisma, *Implementasi Keamanan Web dengan Web Aplikasi Firewall*. Bandung: Unikom, 2013.

[8] STO, *Web Hacking :Skenario & Demo*. Jasakom, 2009.

[9] Pelikaan, Denis. *Naxsi Performance Measurement.* University of Amsterdam, 2013.

[10] F. Setiyawan. *Implementasi Firewall Aplikasi Web untuk Mencegah SQL Injection Menggunakan Naxsi*